



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



İKİLİ AĞIRLIKLİ AĞ VE XNOR-AĞ
İKİLİLEŞTİRİLMİŞ EVRİŞİMSEL DERİN
SİNİR AĞLARININ ANALİZİ

Emir Ali DİNSEL
YÜKSEK LİSANS TEZİ
Bilgisayar Mühendisliği Anabilim Dalı

Ağustos-2020
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Emir Ali DİNSEL tarafından hazırlanan “İkili Ağırlıklı Ağ ve XNOR-Ağ İkileştirilmiş Evrişimsel Derin Sinir Ağlarının Analizi” adlı tez çalışması 21/08/2020 tarihinde aşağıdaki jüri tarafından oy birliği ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

Başkan

Prof. Dr. Harun UĞUZ

Danışman

Doç. Dr. Halife KODAZ

Üye

Dr. Öğr. Üyesi Onur İNAN

İmza

.....

.....

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Saadettin Erhan KESEN
Enstitü Müdürü

ÖZET

YÜKSEK LİSANS

İKİLİ AĞIRLIKLI AĞ VE XNOR-AĞ İKİLİLEŞTİRİLMİŞ EVRİŞİMSEL DERİN SİNİR AĞLARININ ANALİZİ

Emir Ali DİNSEL

Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Halife KODAZ

2020, 107 Sayfa

Jüri

Doç. Dr. Halife KODAZ
Prof. Dr. Harun UĞUZ
Dr. Öğr. Üyesi Onur İNAN

Evrışimsel derin sinir ağları (ESA), çoğunlukla görüntü tanıma ve görüntü tespiti işlemlerinde kullanılmakta ve etkileyici performanslar göstermektedir. ESA modellerinin boyutları gün geçtikçe daha da genişlemektedir. Bu sayede başarıları dolaylı olarak artırmakla beraber yüksek olan parametre sayıları, bellek ve işlem maliyetleri de artmaktadır. Bu yüksek bellek ve işlem maliyetinden dolayı gömülü sistemler, mobil aygıtlar, IoT (Nesnelerin İnterneti) gibi kısıtlı donanımlarda uygulanmasında problemler meydana gelmektedir. Bu problemler için çeşitli çözüm yolları geliştirilmektedir.

ESA'ların yüksek olan bellek ve işlem maliyetlerini azaltmak için ortaya sunulmuş yöntemlerden birisi ağı ikilileştirmektir. ESA'ların büyük bir çoğunluğunun efektif çalışabilmesi için GPU'lar (Grafik İşlem Birimi) gereklidir. Tek başına CPU'lar (Merkezi İşlem Birimi) yeterli olamamaktadır. İkilileştirilmiş yapay derin sinir ağları, GPU bağımlılığına çözüm sunulabilmekte ve orta – düşük performanslı CPU'larda da ESA'ların çalışmasına olanak sağlanabilmektedir.

Bu çalışmada, ağ ikilileştirme yöntemlerinden olan İkili Ağırlıklı Ağ (İAA) ve XNOR-Ağ yapıları incelenmiştir. CIFAR-10, CIFAR-100, MNIST ve SVHN veri setleri kullanılarak LeNet-5, AlexNet, VGG modelleri ve bu çalışma için özelleştirilmiş ağ modelleri ile belirtilen algoritmalar test edilmiştir. Klasik ondalıklı parametrelere sahip ağlarla doğruluk, işlem hızı, bellek maliyeti, enerji tüketimi gibi kriterler temel alınarak karşılaştırmaları yapılmıştır. XNOR-Ağ ile ESA'lar yaklaşık 58 kat daha hızlı çalışmaktadır. Hem XNOR-Ağ hem de İAA ile tek duyarlılık kayan noktalı formata göre 32 kat, çift duyarlılık kayan noktalı formata göre 64 kat bellek tasarrufu sağlanmaktadır.

Anahtar Kelimeler: Evrışimsel Derin Sinir Ağları, Derin Öğrenme, İkili Ağırlıklı Ağ, XNOR-Ağ, İkilileştirilmiş Sinir Ağları, Nicleme, Bellek Kazancı

ABSTRACT

MS THESIS

ANALYSIS OF BINARY WEIGHTED NETWORK AND XNOR-NET BINARIZED CONVOLUTIONAL DEEP NEURAL NETWORKS

Emir Ali DİNSEL

**Konya Technical University
Institute of Graduate Studies
Department of Computer Engineering**

Advisor: Assoc. Prof. Dr. Halife KODAZ

2020, 107 Pages

**Jury
Assoc. Prof. Dr. Halife KODAZ
Prof. Dr. Harun UĞUZ
Asst. Prof. Dr. Öğr. Üyesi Onur İNAN**

Convolutional deep neural networks (CNN) are mostly used in image recognition and image detection processes and show impressive performances. The dimensions of CNN models are expanding day by day. In this way, while relatively increasing the successes, the high number of parameters, memory and transaction costs also increase. Due to this high memory and processing cost, problems occur in the limited hardware such as application of embedded systems, mobile devices, and IoT (Internet of Things). Various solutions are developed for these problems.

One of the methods introduced to reduce the high memory and processing costs of the CNNs is to binarize the network. GPU (Graphic Processing Unit) is required for most of the CNNs to work effectively. Only CPU (Central Processing Unit) is not enough. Binarized artificial deep neural networks can be solved for GPU dependency and make possible to operate the CNNs in mid - low performance CPUs.

In this study, BWN (Binary Weighted Network) and XNOR-NET binarized neural network structures are examined. LeNet-5, AlexNet, VGG models and the models which specialized for this study were tested using CIFAR-10, CIFAR-100, MNIST and SVHN datasets. Comparisons were made with networks with classical decimal parameters based on criteria such as accuracy, processing speed, memory cost and energy consumption. CNNs work approximate 58 times faster with XNOR-NET. With both XNOR-NET and BWN, it is provided with 32 times memory saving compared to single precision floating point format and 64 times memory saving compared double precision floating point format.

Keywords: Convolutional Deep Neural Network, Deep Learning, Binary Weighted Network, XNORNET, Binarized Neural Network, Quantization, Memory Saving

ÖNSÖZ

Bu çalışmamda bana yol gösteren danışman hocam Doç. Dr. Halife KODAZ hocama teşekkür ederim.

Lisans ve lisansüstü eğitimim boyunca değerli bilgi ve tecrübeleri ile bana katkıda bulunan tüm öğretim üyelerine ve öğretim elemanlarına teşekkür ederim.

Çalışmalarım ve eğitim hayatım boyunca destekleriyle beni hiçbir zaman yalnız bırakmayan aileme sonsuz teşekkürler ederim.

Emir Ali DİNSEL
KONYA-2020



İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÖNSÖZ	vi
İÇİNDEKİLER	vii
SİMGELER VE KISALTMALAR.....	ix
1. GİRİŞ	1
1.1. Tezin Organizasyonu.....	2
2. KAYNAK ARAŞTIRMASI	3
2.1. Evrişimsel Sinir Ağları Hakkında Kaynak Araştırması.....	3
2.2. İkili Ağlar Hakkında Kaynak Araştırması	4
3. MATERYAL VE YÖNTEM.....	6
3.1. Yapay Sinir Ağları İçin Temel Kavramlar ve Terminoloji.....	6
3.1.1. Algılayıcı kuralı	6
3.1.2. İleri beslemeli yapay sinir ağları.....	7
3.1.3. Geri yayılım ile çok katmanlı ağların eğitimi.....	8
3.2. Evrişimsel Sinir Ağları	9
3.2.1. Giriş katmanı	10
3.2.2. Evrişim katmanı.....	10
3.2.3. Aktivasyon katmanı	13
3.2.4. Ortaklama katmanı.....	14
3.2.5. Tam bağlantı katmanı.....	16
3.2.6. Seyreltme yöntemi	16
3.2.7. Yığın normalleştirme yöntemi.....	17
3.3. Evrişimsel Ağ Modelleri.....	18
3.3.1. LeNet-5 ağ modeli	18
3.3.2. AlexNet ağ modeli	19
3.3.3. VGG ağ modelleri.....	20
3.3.4. Özelleştirilmiş ağ modelleri	23
3.4. Veri Setleri	26
3.4.1. MNIST veri seti	26
3.4.2. SVHN veri seti.....	28
3.4.3. CIFAR-10 veri seti.....	28
3.4.4. CIFAR-100 veri seti.....	29
3.5. Sinir Ağlarını İkileştirme Yöntemleri	31
3.5.1. İkili ağırlıklı ağ	32
3.5.2. XNOR-Ağ.....	34
3.6. Veri Ön İşleme	37
3.6.1. Veri artırma.....	37

4. DENEYSEL SONUÇLAR.....	38
4.1. MNIST Veri Seti Üzerinde Yapılan Deneyler	38
4.1.1. MNIST veri seti ve LeNet-5 ağ modeli ile yapılan deneyler	38
4.1.2. MNIST veri seti ve VGG-11 ağ modeli ile yapılan deneyler	40
4.1.3. MNIST veri seti ve VGG-19-k ağ modeli ile yapılan deneyler	42
4.2. SVHN Veri Seti Üzerinde Yapılan Deneyler	43
4.2.1. SVHN veri seti ve LeNet-5 ağ modeli ile yapılan deneyler	44
4.2.2. SVHN veri seti ve VGG-11 ağ modeli ile yapılan deneyler	45
4.2.3. SVHN veri seti ve VGG-19-k ağ modeli ile yapılan deneyler	47
4.3. CIFAR-10 Veri Seti Üzerinde Yapılan Deneyler	49
4.3.1. CIFAR-10 veri seti ve LeNet-5 ağ modeli ile yapılan deneyler	49
4.3.2. CIFAR-10 veri seti ve AlexNet ağ modeli ile yapılan deneyler	51
4.3.3. CIFAR-10 veri seti ve VGG-11 ağ modeli ile yapılan deneyler	53
4.3.4. CIFAR-10 veri seti ve VGG-13 ağ modeli ile yapılan deneyler	57
4.3.5. CIFAR-10 veri seti ve VGG-16 ağ modeli ile yapılan deneyler	58
4.3.6. CIFAR-10 veri seti ve VGG-19 ağ modeli ile yapılan deneyler	60
4.3.7. CIFAR-10 veri seti ve AlexNet-k ağ modeli ile yapılan deneyler	62
4.3.8. CIFAR-10 veri seti ve VGG-11-k ağ modeli ile yapılan deneyler	64
4.3.9. CIFAR-10 veri seti ve VGG-13-k ağ modeli ile yapılan deneyler	66
4.3.10. CIFAR-10 veri seti ve VGG-16-k ağ modeli ile yapılan deneyler	68
4.3.11. CIFAR-10 veri seti ve VGG-19-k ağ modeli ile yapılan deneyler	70
4.4. CIFAR-100 Veri Seti Üzerinde Yapılan Deneyler	74
4.4.1. CIFAR-100 veri seti ve AlexNet ağ modeli ile yapılan deneyler	74
4.4.2. CIFAR-100 veri seti ve VGG-11 ağ modeli ile yapılan deneyler	76
4.4.3. CIFAR-100 veri seti ve VGG-13 ağ modeli ile yapılan deneyler	78
4.4.4. CIFAR-100 veri seti ve VGG-16 ağ modeli ile yapılan deneyler	80
4.4.5. CIFAR-100 veri seti ve VGG-19 ağ modeli ile yapılan deneyler	82
4.4.6. CIFAR-100 veri seti ve AlexNet-k ağ modeli ile yapılan deneyler	84
4.4.7. CIFAR-100 veri seti ve VGG-11-k ağ modeli ile yapılan deneyler	86
4.4.8. CIFAR-100 veri seti ve VGG-13-k ağ modeli ile yapılan deneyler	87
4.4.9. CIFAR-100 veri seti ve VGG-16-k ağ modeli ile yapılan deneyler	89
4.4.10. CIFAR-100 veri seti ve VGG-19-k ağ modeli ile yapılan deneyler	91
5. SONUÇLAR VE ÖNERİLER.....	93
5.1. Veri Artırma Yapılmayan Deneylere Ait Sonuçlar	93
5.2. Veri Artırma Yapılan Deneylere Ait Sonuçlar	96
5.3. Öneriler	103
KAYNAKLAR	105
ÖZGEÇMİŞ.....	107

SİMGELER VE KISALTMALAR

Simgeler

$\hat{x}$	:	normalleştirilmiş küme
x_i	:	normalleştirilen küme
μ_β	:	normalleştirilen kümenin ortalaması
σ_β^2	:	normalleştirilen kümenin varyansı
A	:	adım sayısı
C	:	hata
D_1	:	giriş görüntü boyutunun derinlik değeri
Epok	:	tüm örneklerin birer kere ağa girdi olarak verilmesi
F	:	filtre uzunluğu
F	:	filtre boyutu
G_1	:	giriş görüntü boyutunun genişlik değeri
I	:	girdi büyüklüğü
lr	:	öğrenme katsayısı
MB	:	megabayt
N	:	iterasyon sayısı
$P_{\text{başlangıç}}$	:	doldurma başlangıcı
P_{son}	:	doldurma bitişi
S	:	adım aralığı
SGD	:	stokastik gradyan inişi
W	:	standart ondalıklı ağırlık
W^B	:	ikili ağırlık
X	:	giriş görüntüsü
Y_1	:	giriş görüntü boyutunun yükseklik değeri
O	:	çıkış büyüklüğü
$Sign(x)$	:	Signum fonksiyonu
b	:	bias
dW	:	ağırlık değişimi için hesaplanan eğim
w_i	:	ağ ağırlıkları
x_i	:	nöron girişleri
I	:	giriş değerleri
W	:	ondalıklı standart ağırlıklar
$\odot$	:	“özel veya değil” işlemi
$\oplus$	:	çarpma işlemi içermeyen evrişim işlemi
B	:	ikili ağırlıklar
n	:	ağırlıkların sayısı
α	:	optimum ölçekleme çarpanı

Kısaltmalar

AlexNet	:	8 katmanlı bir ESA modeli
AlexNet-k	:	AlexNet ESA modelinden türetilen özel bir ağ modeli
BNN	:	İkili Sinir Ağı
CPU	:	Merkezi İşlem Birimi
DSA	:	Derin Sinir Ağı
Epok	:	Tüm örneklerin birer kez girdi olarak kullanıldığı dönem

ESA	:	Evrişimsel Sinir Ağı
Evrişim3-128	:	128x3x3 filtre uygulanan evrişim işlemi
Evrişim3-256	:	256x3x3 filtre uygulanan evrişim işlemi
Evrişim3-384	:	384x3x3 filtre uygulanan evrişim işlemi
Evrişim3-512	:	512x3x3 filtre uygulanan evrişim işlemi
Evrişim3-64	:	64x3x3 filtre uygulanan evrişim işlemi
Evrişim5-256	:	256x5x5 filtre uygulanan evrişim işlemi
Evrişim11-96	:	96x11x11 filtre uygulanan evrişim işlemi
GPU	:	Grafik İşlem Birimi
IoT	:	Nesnelerin İnterneti
İAA	:	İkili Ağırlıklı Ağ
LeNet-5	:	5 katmanlı bir ESA modeli
MNIST	:	Modified National Institute of Standards and Technology
ReLU	:	Düzeltilmiş Doğrusal Birim
Signum	:	Sign
Tanh	:	Hiperbolik Tanjant
VGG-11	:	11 katmanlı bir ESA modeli
VGG-11-k	:	VGG-11 ESA modelinden türetilen özel bir ağ modeli
VGG-13	:	13 katmanlı bir ESA modeli
VGG-13-k	:	VGG-13 ESA modelinden türetilen özel bir ağ modeli
VGG-16	:	16 katmanlı bir ESA modeli
VGG-16-k	:	VGG-16 ESA modelinden türetilen özel bir ağ modeli
VGG-19	:	19 katmanlı bir ESA modeli
VGG-19-k	:	VGG-19 ESA modelinden türetilen özel bir ağ modeli
XNOR-Ağ	:	Özel veya Değil Ağı
YSA	:	Yapay Sinir Ağı

1. GİRİŞ

Derin sinir ağları (DSA), bilgisayarlı görü ve konuşma tanıma da dâhil olmak üzere çeşitli uygulama alanlarında önemli gelişmeler göstermiştir. DSA'ların türlerinden biri olan ve Evrişimsel Sinir Ağları (ESA) olarak bilinen algoritmalar, nesne tanıma ve algılama gibi bilgisayarlı görü problemlerinde son teknoloji sonuçları göstermiştir (Rastegari ve ark., 2016).

ESA'lar, göstermiş olduğu başarının yanında yüksek bellek ve işlem maliyeti gibi sorunları da beraberinde getirmektedir. ESA modelleri büyük ve derin yapıları nedeniyle çok sayıda parametreye sahiptir. Yüksek sayıda olan parametre sayıları nedeniyle yüksek bellek maliyeti oluşur. Aynı zamanda, bu çok sayıda parametre kullanılarak sonuç elde edilmeye çalışılırken birçok matris çarpımı işlemi gerçekleştirilir. Tüm bunlar yüksek işlem maliyetine de sebep olur. İşlem maliyeti o kadar yüksektir ki, bu ağların büyük bir çoğunluğu Merkezi İşlem Birimleri (CPU) üzerinde çalışamazlar. Bu yüksek işlem gücünü karşılamak için orta-yüksek seviye Grafik İşlem Birimleri (GPU) kullanılır (Krizhevsky ve ark., 2012).

Yüksek bellek ve işlem maliyetini donanımsal olarak karşılamak, oldukça pahalı sistemlere ihtiyaç duyulmasına sebebiyet vermektedir. Diğer bir sorun ise bu çözümlerin fiziksel olarak çok fazla yer kaplamalarına ve taşınabilir olma limitlerini zorlayan ağırlıklara ve hacimlere sahip olmalarına sebebiyet vermektedir. Günümüzde taşınabilir cihazlar, giyilebilir teknolojiler, nesnelerin interneti (IoT), gömülü sistemler, mobil aygıtlar gibi birçok donanım ESA'ları gerçek zamanlı uygulamalarda çalıştırmak için gerek finansal gerek taşınabilir olma kısıtlarına takılabilmektedirler. Bu kısıtları azaltmak ve GPU gerekliliğini sona erdirip ESA'ların CPU'lar üzerinde de çalışmasına olanak sağlayacak olan çözümlerden biri de ESA'ları ikilileştirmektir (Courbariaux ve ark., 2015).

İkilileştirilmiş ağlar üzerine sunulmuş algoritmalarından olan İkili Ağırlıklı Ağ (İAA), ağırlıkların ikili değer alacak şekilde eğitilerek bellek maliyetini, tek duyarlılık kayan noktalı formata göre 32 kat, çift duyarlılık kayan noktalı formata göre 64 kat azaltmakla beraber işlem maliyetini de 2 kat azaltmaktadır (Rastegari ve ark., 2016).

İkilileştirilmiş ağlar üzerine sunulmuş başka bir algoritma da Özel veya Değil Ağıdır (XNOR-Ağ). Ağırlıkların ikili değer alacak şekilde eğitilerek bellek maliyetini, tek duyarlılık kayan noktalı formata göre 32 kat, çift duyarlılık kayan noktalı

formata göre 64 kat azaltır. Bununla birlikte ağ eğitilirken her katman öncesi girişler Signum (Sign) fonksiyonu kullanılarak ikili değere indirgenir. Girişlerin ve ağırlıkların ikili olması sayesinde çarpma işlemi yerine ‘özel veya değil’ işlemi kullanılarak matrislerden sonuçlar elde edilir. Bu sayede işlem maliyeti 58 kata kadar azaltılabilmekte ve ESA’ların CPU’lar üzerinde çalışabilmesi sağlanabilmektedir (Rastegari ve ark., 2016)(Hubara ve ark 2016).

Yapılan tez çalışması kapsamında, Yapay Sinir Ağlarını (YSA) ikilileştirmek için kullanılan İAA ve XNOR-Ağ yöntemleri çeşitli ESA mimarilerine uygulanmıştır. Bu uygulamalarda çeşitli veri setleri üzerinden sınıflandırma işlemi gerçekleştirilmiştir.

MNIST, SVHN ve CIFAR-10 veri setleri üzerinde LeNet-5, VGG-11 ve VGG-19-k ESA modelleri kullanılarak ikilileştirme yöntemleri test edilmiştir.

CIFAR-10 ve CIFAR-100 veri setleri üzerinde AlexNet, VGG ağ modelleri ve bu tez çalışması için özelleştirdiğimiz ESA modelleri kullanılarak ikilileştirme yöntemleri test edilmiştir.

Tüm bu çalışmalar sonucu elde edilen sınıflandırma doğruluğu, bellek maliyeti, işlem maliyeti ve enerji maliyeti kıstasları üzerinden bu iki metodun ve standart ESA’ların karşılaştırmaları ve değerlendirmeleri yapılmıştır.

1.1. Tezin Organizasyonu

Bölüm 1, tezin yapısı ve temel amacı hakkında kısa bilgi vermektedir.

Bölüm 2, literatür araştırmalarını içermektedir.

Bölüm 3, bu tez çalışmasında kullanılan veriler, materyaller ve yöntemler hakkında bilgileri içermektedir.

Bölüm 4, Ağ modelleri ikilileştirilerek, standart yapıları ve ikilileştirilmiş yapıları çeşitli veri setleri üzerinden karşılaştırılmıştır.

Bölüm 5, Yapılmış olan deneylere ait sonuçları ve önerileri içermektedir.

2. KAYNAK ARAŞTIRMASI

2.1. Evrişimsel Sinir Ağları Hakkında Kaynak Araştırması

Literatürde bilinen ilk Evrişimsel Yapay Derin Sinir Ağı, Yann Lecun ve arkadaşları tarafından 1998 yılında duyurulmuştur. LeNet-5 olarak adlandırılan bu ağın giriş katmanı 32x32 boyutundadır. İki adet evrişim ve iki adet havuzlama katmanına sahip olan ağın çıkış boyutu 10' dur. Çalışmanın yapıldığı yıllarda, ESA'ların ihtiyaç duyduğu büyük verilerin elde edilememesinden ve yeterli donanım gücünün sağlanamamasından dolayı uzun yıllar ESA'lar büyük gelişmeler gösterememiştir(LeCun ve ark 1998).

2012 yılında ESA modellerinin ve derin öğrenmenin tekrar popüler hale gelmesini sağlayan ilk çalışmadır. Alex Krizhevsky, Ilya Sutskever ve Geoffrey Hinton tarafından geliştirilmiştir. Birbirini takip eden evrişim ve havuzlama katmanları bulunmasından dolayı temel olarak LeNet-5 modeline çok benzemektedir. Aktivasyon fonksiyonu olarak Düzeltilmiş Doğrusal Birim (ReLU), ortaklama katmanlarında da maksimum ortaklama kullanılmaktadır. Daha büyük ve daha derin olan bu ağ modeli paralel çift GPU üzerinde çalıştırılmıştır. Yaklaşık olarak 60 milyon parametre hesaplanması yapılmaktadır. ImageNet ILSVRC yarışmasında Top-5 sınıflandırma doğruluk oranını %74,3'ten %82,7'ye çıkararak, ani bir yükselme sağlamıştır. Görüntü sınıflandırma probleminde bir kırılma noktası denilebilir(Krizhevsky ve ark 2012).

2012 yılında AlexNet'in ImageNet yarışmasını kazanmasının akabinde yapılan yarışmalarda derin öğrenme modelleri kullanılmaya başlandı. Matthew Zeiler ve Rob Fergus'un tasarlamış olduğu ZFNet, 2013 yılında ImageNet yarışmasının kazananı olmuştur. Bu model ile nesne tanımda hata oranı %11,2'ye indirilmiştir. Bu mimari AlexNet mimarisinin geliştirilmiş halidir(Zeiler ve Fergus 2014).

Simonyan ve Zisserman tarafından yapılan çalışmada, Evrişimsel Ağ derinliğinin büyük ölçekli görüntü tanıma ayarındaki doğruluğu üzerindeki etkisi araştırılmıştır. Çalışmanın ana katkısı, çok küçük (3x3) kıvrımlı filtreler sahip bir mimari kullanarak artan derinlik ağlarının kapsamlı bir değerlendirmesidir. Derinliğin 16-19 ağırlık katmanlarına itilmesiyle önemli bir iyileşmenin sağlanabileceğini göstermektedir. Bu bulgular, ekibi birinci ve ikinci sırayı aldığı ImageNet 2014 yarışması sunumunun temelini oluşturur. En iyi performans gösteren iki ESA modeli, bilgisayarlı görme üzerine derin görsel sunumların kullanımı hakkında daha fazla araştırmayı kolaylaştırmak için halka açık hale getirilmiştir (Simonyan ve Zisserman 2014).

GoogLeNet yapısındaki iç içe geçen modüllerinden dolayı karmaşık bir mimaridir. GoogLeNet 22 katmanlı modeli %5,7 hata oranı ile ImageNet 2014 yarışmasının kazananı olmuştur. Bu mimari genel olarak, ardışık bir yapıda evrişim ve havuzlama katmanlarını üst üste istiflemekten uzaklaşan ilk ESA mimarilerinden birisidir. Ayrıca bu yeni model bellek ve güç kullanımı üzerinde önemli bir yere sahiptir. Çünkü katmanların hepsini yığınlamak ve çok sayıda filtre eklemek, bir hesaplama ve bellek maliyeti getirir ve ezberleme olasılığını artırır. GoogLeNet mimarisinde bu durumun üstesinden gelmek için paralel olarak birbirine bağlı modüller kullanılmıştır (Szegedy ve ark 2015).

ResNet, kendisinden önceki tüm mimarilerden daha derin olarak tasarlanan bir mimaridir. 152 katmandan oluşmaktadır. Aynı zamanda %3,6 (Becerileri ve uzmanlıklarına bağlı olarak, insanlar genelde %5-10 hata oranına sahiptirler) hata oranı ile ImageNet 2015 yarışmasının kazananı olmuştur. Bu mimari özel bloklardan oluşmaktadır (He ve ark 2016).

Özkan ve Ülker'in yaptığı bu çalışmada Derin Öğrenme hakkında detaylı bilgiler verilmiştir. ESA mimarisinin katmanları olan Evrişim, Havuzlama, ReLU, Seyreltme, Tam bağlantılı ve Sınıflandırma katmanı hakkında açıklamalar yapılmıştır (Özkan ve Ülker 2017).

2.2. İkili Ağlar Hakkında Kaynak Araştırması

Derin Öğrenme için özel donanımların araştırılması ve geliştirilmesine büyük ilgi duyulmaktadır. İkili ağırlıklar, yani yalnızca iki olası değerle (örneğin, -1 veya 1) kısıtlanan ağırlıklar, çarpma işlemini basit hesaplamalar ile gerçekleştirebilen özel derin öğrenme donanımlarının tasarlanmasında büyük faydalar sağlayacaktır. Courbariaux ve arkadaşları tarafından yapılan çalışmada, ağ ikilileştirme yöntemlerinden olan BinaryConnect sunulmuştur. MNIST, CIFAR-10 ve SVHN veri kümelerinde BinaryConnect ile son teknoloji sonuçları elde edilmiştir (Courbariaux ve ark 2015).

Hubara ve arkadaşları tarafından yapılan çalışmada, ikili ağırlıklara ve çalışma zamanında ikili aktivasyona sahip olan İkili Sinir Ağlarını (BNN) eğitmek için bir yöntem sunulmuştur. Bu yöntemde, eğitim zamanında, ikili eğimleri ve aktivasyonları hesaplamak için parametre eğimleri kullanılır. İleri besleme sırasında, BNN'ler bellek boyutunu ve erişimlerini büyük ölçüde azaltır ve çoğu aritmetik işlemi, güç verimliliğini önemli ölçüde iyileştirmesi beklenen ikili işlemlerle değiştirir. BNN'lerin etkinliğini

doğrulamak için Torch7 ve Theano yapıları üzerinde iki takım deney yapılmıştır. Her ikisinde de BNN'ler neredeyse üst düzey sonuçlar elde etmiştir. Son olarak, sınıflandırma doğruluğunda herhangi bir kayıp yaşamadan, MNIST BNN'ini, optimize edilmemiş bir GPU çekirdeğinden 7 kat daha hızlı çalıştırmanın mümkün olduğu ikili bir matris çarpma GPU çekirdeği yazılmıştır (Hubara ve ark 2016).

Rastegari ve arkadaşları tarafından yapılan çalışmada, standart ESA'lara, İkili Ağırlıklı Ağlar (İAA) ve XNOR-Ağlar olmak üzere iki etkin yaklaşım önerilmiştir. İAA'larda, parametreler ikili değerler aldığı için yaklaşık olarak 32 kat bellek tasarrufu sağlanmıştır. XNOR-Ağ'da hem filtreler hem de evrişim katmanlarına girdi ikilidir. Bu sayede, yaklaşık olarak 58 kat daha hızlı evrişim işlemleri ve 32 kat bellek tasarrufu sağlanmıştır. XNOR-Ağ, CPU'larda (GPU'lar yerine) en yeni ağları gerçek zamanlı olarak çalıştırma imkânı sunar. İkili ağlar basit, doğru, verimli ve zorlu görsel görevler üzerinde çalışır. Sunulan yöntem, ağ ikilileştirme yöntemlerinden olan BinaryConnect ve BinaryNet ile karşılaştırılmış ve bu yöntemleri ImageNet' teki büyük marjlarla, Top-1'de % 16'dan üzerinde bir başarıyla geçmiştir (Rastegari ve ark 2016).

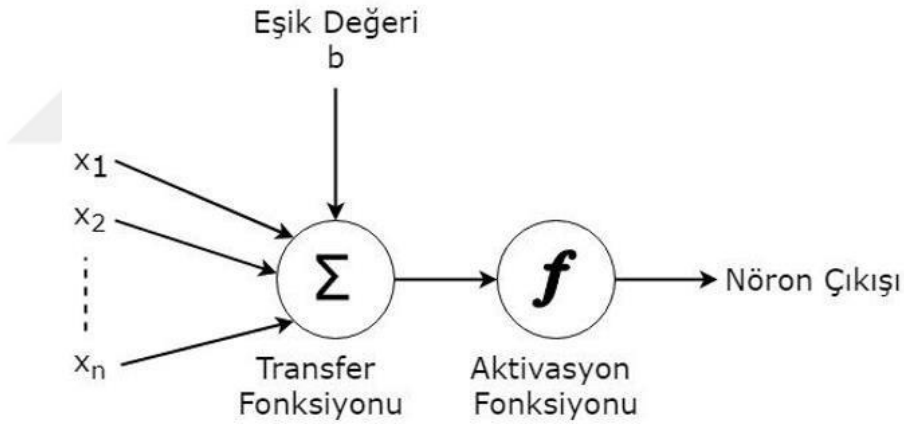
Sun ve arkadaşları tarafından yapılan çalışmada, eşdeğer XNOR ve bit sayma işlemini paralel bir şekilde uygulayan bit-hücre tasarımına sahip bir RRAM sinaptik mimarisi (XNOR-RRAM) önermiştir. RRAM modelleri ve 65nm CMOS PDK ile 128×128 alt dizi boyutu ve 3 bit MLSA içeren sistem ile deneyler yapmışlardır. Yapılan deneylerde MNIST veri seti kullanılan çok katmanlı yapay sinir ağında %98,43 ve CIFAR-10 veri seti kullanılan ESA'larda %86,08 doğrulama değerleri elde edilmiştir. XNOR-RRAM'ın öngörülen enerji verimliliği 141.18 TOPS / W' dır ve sıralı satır satır okumaya sahip geleneksel RRAM sinaptik mimarisine kıyasla yaklaşık 33 kat gelişme göstermiştir (Sun ve ark 2018).

3. MATERYAL VE YÖNTEM

Bu bölümde tez için kullanılacak olan algoritmalar, veri kümeleri, ağ mimarileri, ağ ikilileştirme yöntemleri ve ESA modelleri hakkında bilgiler verilmiştir.

3.1. Yapay Sinir Ağları İçin Temel Kavramlar ve Terminoloji

Geri yayılım, derin öğrenme için temel öğrenme yöntemidir. Ancak geri yayılımı araştırmaya başlamadan önce, bir dizi temel kavramı tanımlamalı ve etkileşimleri açıklanmalıdır. Derin öğrenme, derin yapay sinir ağları ile uygulanan makine öğrenmesi algoritmalarını içermektedir. Sığ sinir ağları basit ileri beslemeli sinir ağları olarak da tanımlanabilir, ancak terimin kendisi sadece sığ olanları değil, geri bildirim bağlantısı olmayan herhangi bir sinir ağını da kapsamaktadır. Bu anlamda, ESA aynı zamanda ileri beslemeli bir sinir ağıdır, ancak sığ bir sinir ağı değildir. Genel olarak, derin öğrenme, sığ bir sinir ağına daha fazla katman eklemeye çalıştığımızda ortaya çıkan sorunları tespit etmekten oluşur (Skansi 2018). Yapay nöron modellemesi Şekil 3.1’de gösterilmiştir.



Şekil 3.1. Yapay Nöron (Yacim ve Boshoff 2018)

3.1.1. Algılayıcı kuralı

Nöronlardaki öğrenme süreci, geri yayılım eğitimi sırasında ağırlıkların ve bias değerlerinin modifikasyonu veya güncellenmesidir. Sınıflandırma sırasında sadece ileri geçiş yapılır. Yapay nöronlar için erken öğrenme prosedürlerinden biri de algılayıcı öğrenmedir. Algılayıcı, ikili eşikli bir nöron ve algılayıcı öğrenme kuralından oluşur ve modifiye edilmiş bir lojistik regresyona benzer (Skansi 2018). İkili eşik nöronun tanımlaması Denklem 3.1 ve 3.2 ’de verilmiştir.

$$z = b + \sum_i w_i x_i \quad (3.1)$$

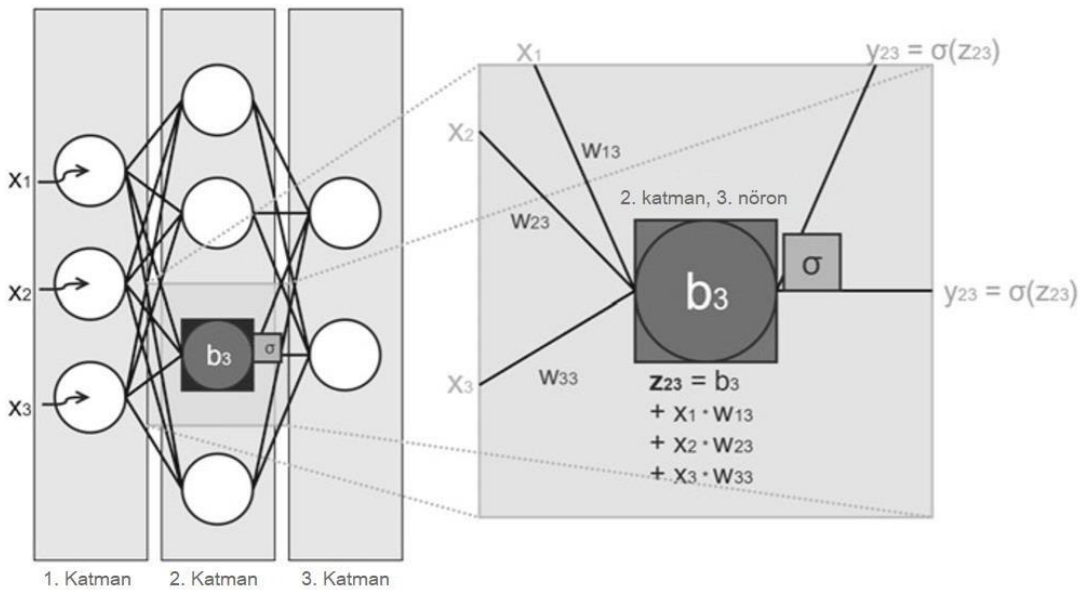
$$y = \begin{cases} 1, & z \geq 0 \\ 0, & \text{değilse} \end{cases} \quad (3.2)$$

Burada x_i girişleri, w_i ağırlıkları, b biası temsil eder. Denklem 3.2 genellikle doğrusal olmama ile yapılan kararı tanımlar, ancak burada bunun yerine ikili bir adım fonksiyonu kullanılabilir (Skansi 2018).

3.1.2. İleri beslemeli yapay sinir ağları

Genellikle ileri beslemeli sinir ağları veya çok katmanlı algılayıcılar (MLP) olarak da adlandırılan derin ileri beslemeli ağlar, en özünde derin öğrenme modelleridir. İleri beslemeli bir ağın amacı, f^* fonksiyonuna yaklaşımdır. Örneğin, bir sınıflandırıcı için $y = f^*(x)$ haritası ve bir x girişi bir y kategorisiyle eşleştirilsin. İleri beslemeli bir ağ, bir $y = f(x; \theta)$ eşlemesini tanımlar ve en iyi sonuca yaklaştıran θ parametrelerinin değerini öğrenir. Bu modellere ileri besleme denir, çünkü bilgi girişlerden başlayarak ilerler, ara hesaplardan da geçtikten sonra son olarak y çıkışına kadar gelir (Goodfellow ve ark 2016).

Şekil 3.2’de basit bir üç katmanlı sinir ağının yapısı gösterilmiştir. Bir katmanın her nöronu bir sonraki katmanın tüm nöronlarına bağlanır ve bir önceki katmanın miktarının ne kadarının bir sonraki katmanın belirli bir nöronuna aktarılacağını belirleyen ilgili ağırlıkla çarpılır.

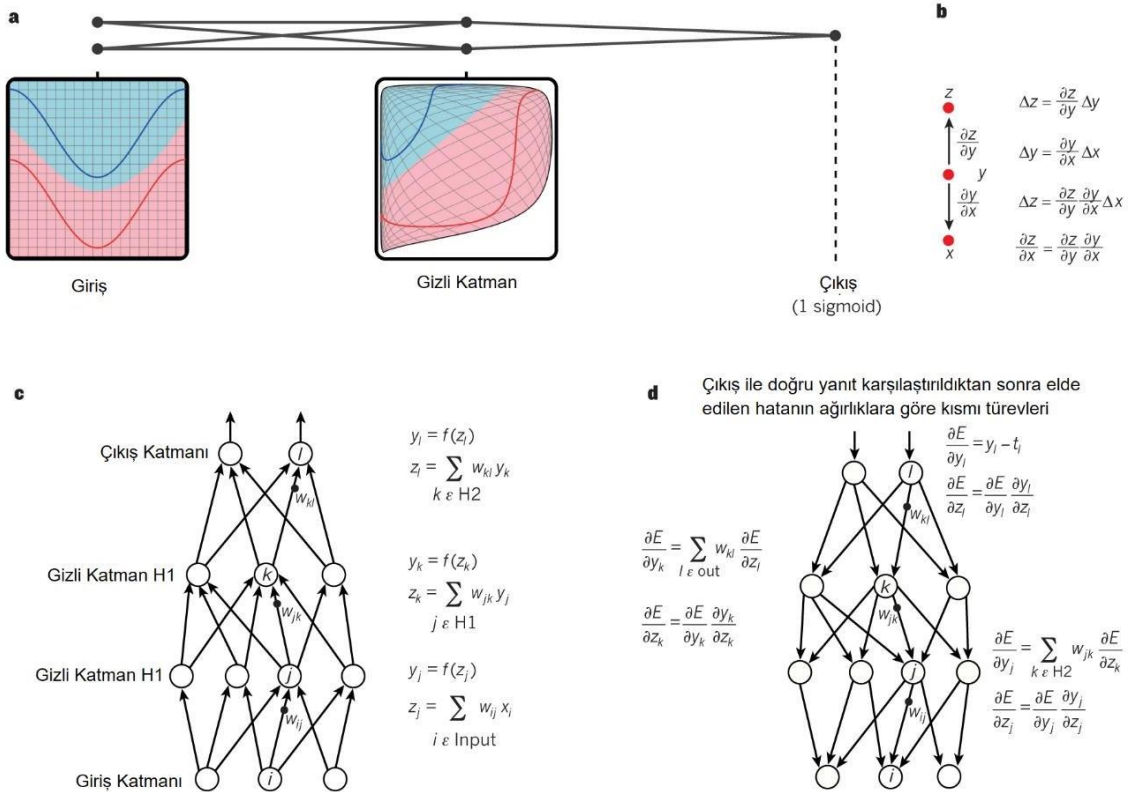


Şekil 3.2. Üç katmanlı YSA(Skansi 2018)

3.1.3. Geri yayılım ile çok katmanlı ağların eğitimi

Örüntü tanımanın ilk günlerinden itibaren araştırmacıların amacı, el yapımı özellikleri eğitilebilir çok katmanlı ağlarla değiştirmektir, ancak basitliğine rağmen, çözüm 1980'lerin ortalarına kadar yaygın olarak anlaşılmamıştır. Çok katmanlı mimariler basit azalan eğim ile eğitilebilir. Modüller, girdilerinin ve iç ağırlıklarının nispeten düzgün işlevleri olduğu sürece, geri yayılım prosedürü kullanılarak eğimler hesaplanabilir. Bunun yapılabileceği ve işe yaradığı fikri, 1970 ve 1980'lerde birkaç farklı grup tarafından bağımsız olarak keşfedilmiştir (LeCun ve ark 2015).

Çok katmanlı bir modül yığınının ağırlıklarına göre objektif bir fonksiyonun eğimini hesaplamak için geri yayılım prosedürü, türevleri için zincir kuralının pratik bir uygulamasından başka bir şey değildir. Temel görüş, bir modülün girişine göre hedefin türevinin, bu modülün çıkışına (veya sonraki modülün girişine) göre eğimin geriye doğru çalışarak hesaplanabilmesidir. Şekil 3.3'de görüldüğü üzere geri yayılım denklemi, eğimleri üstteki çıkıştan (ağın tahminini ürettiği yerde) en alta kadar (harici girişin beslendiği yere) ilerleyerek tüm modüller boyunca yaymak için tekrar tekrar uygulanabilir. Bu eğim değerleri hesaplandıktan sonra, her bir modülün ağırlıklarına göre değişim miktarları hesaplanabilir (LeCun ve ark 2015).

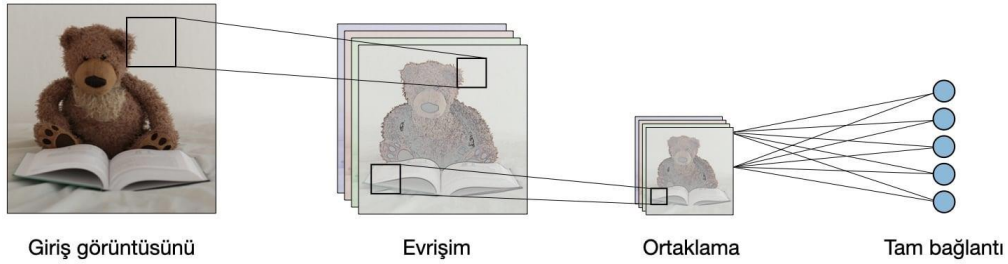


Şekil 3.3. Çok katmanlı YSA ve geri yayılım hesaplaması (LeCun ve ark 2015)

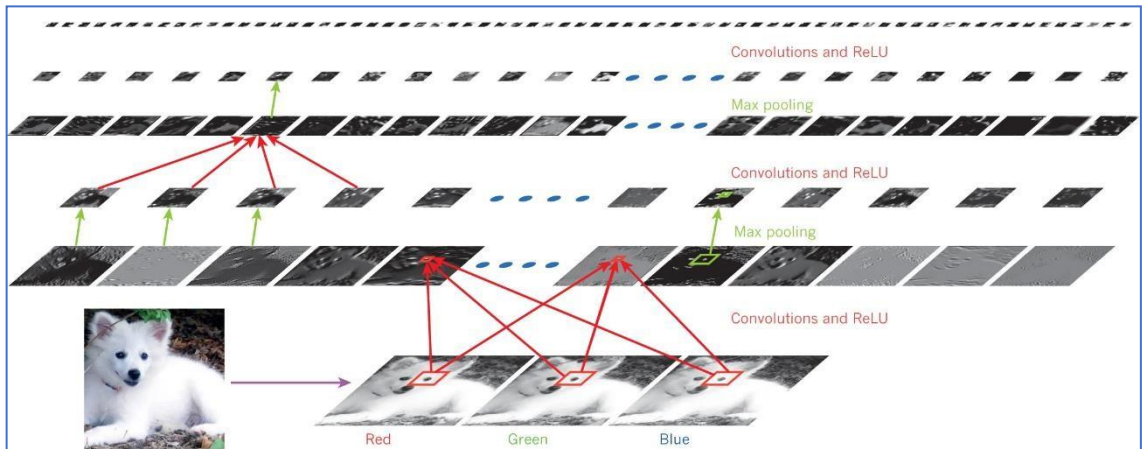
3.2. Evrişimsel Sinir Ağları

LeCun tarafından 1989 yılında geliştirilen ESA'lar, görüntüler gibi ızgara benzeri topolojiye sahip verileri işleyip sonuçlar üreten özel bir tür sinir ağıdır. Evrişimli ağlar pratik uygulamalarda son derece başarılı olmuştur. ESA'lar adını evrişim adı verilen bir matematik işleminden alır. Evrişim özel bir doğrusal işlem türüdür. Evrişimli ağlar, katmanlarından en az birinde genel matris çarpımı yerine evrişim işlemini kullanan yapay sinir ağlardır (Goodfellow ve ark 2016).

Geleneksel ESA mimarisi Şekil 3.4 ve Şekil 3.5'de gösterilen bölümlerden oluşur. Bu bölümler, giriş katmanı, evrişim katmanı, ortaklama katmanı, aktivasyon katmanı ve tam bağlantı katmanıdır. Şekil 3.4'de aktivasyon fonksiyonu olarak ReLU kullanılmıştır. Şekil 3.5'de köpek görüntüsü (sol alt; RGB) (kırmızı, yeşil, mavi) giriş olarak verilmiştir. Tipik ESA mimarisinin her katmanının (filtreleri değil) çıkışları öğrenilen özelliklerden birinin çıktısına karşılık gelen ve görüntü konumlarının her birinde algılanan bir özellik haritasıdır. Bilgi aşağıdan yukarıya doğru akar, alt düzey özellikler yönlendirilmiş kenar algılayıcıları olarak işlev görür ve çıktındaki her görüntü sınıfı için bir puan hesaplanır (LeCun ve ark 2015).



Şekil 3.4. Geleneksel ESA yapısı (Amidi ve Amidi 2018)



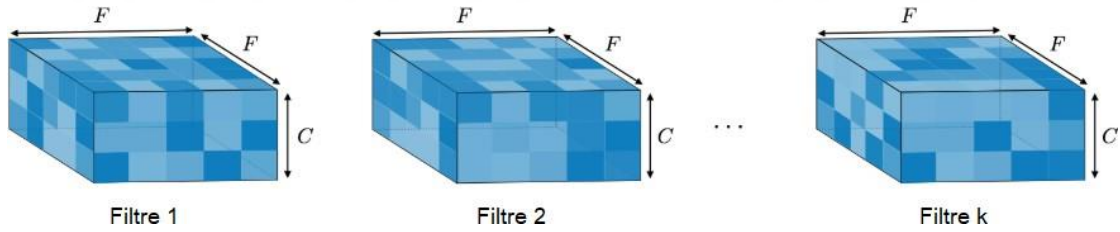
Şekil 3.5. Tipik ESA katmanları ve yapısı (LeCun ve ark 2015)

3.2.1. Giriş katmanı

ESA modellerinin ilk katmanıdır. Genellikle girdi olarak görüntü vb. matris yapısında veriler sağlanmaktadır. Sınıflandırma işlemlerinde ağın başarısı için girdi boyutu önem arz etmektedir. Girdi boyutunun büyümesi ağ performansını artırmakla beraber işlem ve bellek maliyetleri de artmış olur. Bu yüzden girdi verisinin boyutu tüm bu kıstaslar hesaplanarak amacına uygun olarak seçilmelidir (Özkan ve Ülker 2017).

3.2.2. Evrişim katmanı

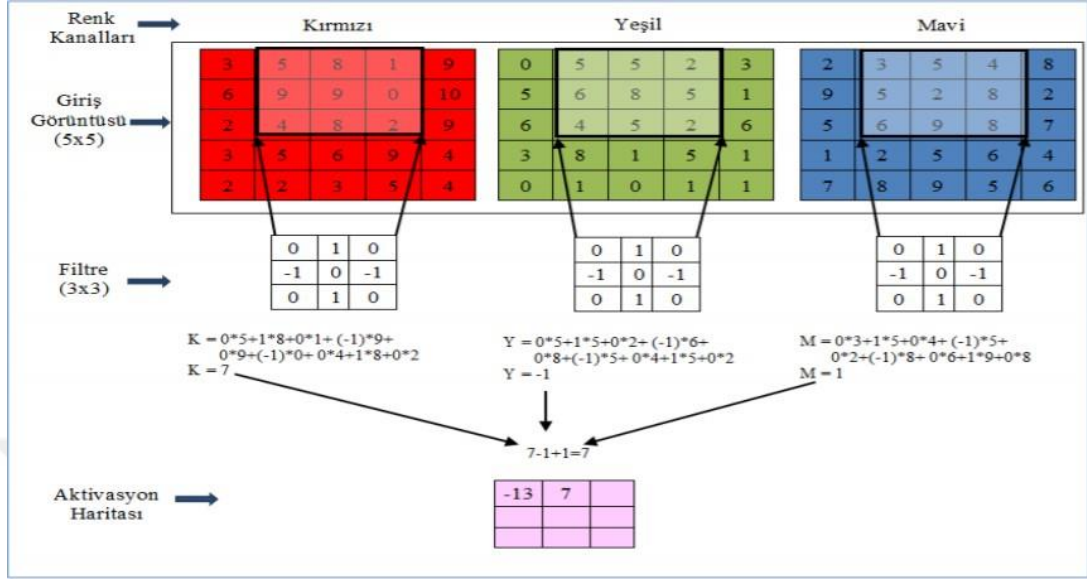
Evrişim, bir fonksiyonun şeklinin, başka bir fonksiyon tarafından nasıl değiştirildiğini ifade eden üçüncü bir fonksiyon üreten matematiksel bir işlemdir. Evrişim katmanında çeşitli en, boy ve genişlikteki filtreler görüntü vb. matrislerin üzerinde gezdirilerek çarpma ve toplama işlemlerinden geçirilir. Sonuç yine matris ya da çok boyutlu bir matristir. Evrişim katmanı, evrişim işlemlerini gerçekleştiren filtreleri, 'I' girişini boyutlarına göre tararken kullanır. Hiper parametreleri olarak 'F' filtre boyutunu temsil eder. Filtreler Şekil 3.6'da görüldüğü gibi en, boy ve yükseklik değerlerinden oluşur (Amidi ve Amidi 2018).



Şekil 3.6. Filtre görselleri (Amidi ve Amidi 2018)

Elde edilen çıktı, öznitelik haritası veya aktivasyon haritası olarak adlandırılır. Filtreler 3x3, 5x5, 11x11 gibi farklı boyutlarda olabilir. Filtreler, bir önceki katmandan gelen tensörlere evrişim işlemini uygulayarak çıkış verisini oluştururlar. Şekil 3.7'de evrişim katmanına ait örnek görsel sunulmuştur. Bu görselde görüldüğü gibi filtre derinliği boyutu girdi derinliği boyutuna eşittir. Derinlik boyutu 3 olan RGB (Kırmızı-Yeşil-Mavi) formatında görüntüye yine derinlik boyutu 3 olan 5x5 boyutlarında filtre uygulanmıştır. Filtre değerleri, örtüştüğü girdi değerleri ile çarpılır ve tüm bu çarpımlar toplanarak çıktı matrisi için bir değer oluşturur. 3x5x5 girdiye 3x3x3 boyutunda filtre adım değeri 1 olarak uygulandığında çıktı olarak 1x3x3 boyutlu bir matris oluşur. Bu

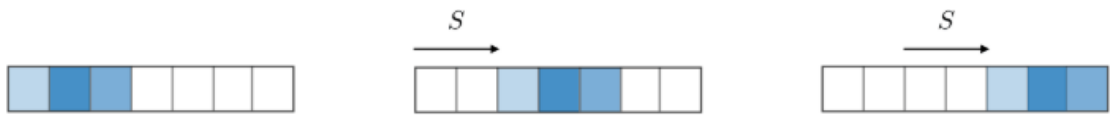
örnekte doldurma (padding) işlemi uygulanmamıştır (LeCun ve ark 2015, Özkan ve Ülker 2017).



Şekil 3.7. 3x5x5 boyutlu girişe 3x3x3 boyutunda filtre uygulandığında evrişim işlemi (Özkan ve Ülker 2017)

3.2.2.1. Adım aralığı ve doldurma

Adım aralığı, evrişim veya bir ortaklama işlemi için, S adımı (adım aralığı), Şekil 3.8’de görüldüğü gibi her işlemden sonra filtrenin hareket ettiği piksel sayısını belirtir. Adım değeri evrişim katmanında genellikle 1 seçilirken, ortaklama katmanında 2 değeri verilir. Ancak yüksek boyutlu girdilerin işlem maliyetini düşürmek için adım değeri özellikle ilk katmanlarda 11 gibi 7 gibi yüksek değerlerde olabilir. Bu sayede çıktının boyutu azalır ama teorik olarak veri kaybı da artmış olur (Krizhevsky ve ark 2012, Amidi ve Amidi 2018).



Şekil 3.8. Adım değeri 2 olduğunda filtrenin işlem yaptığı bölgeler (Amidi ve Amidi 2018)

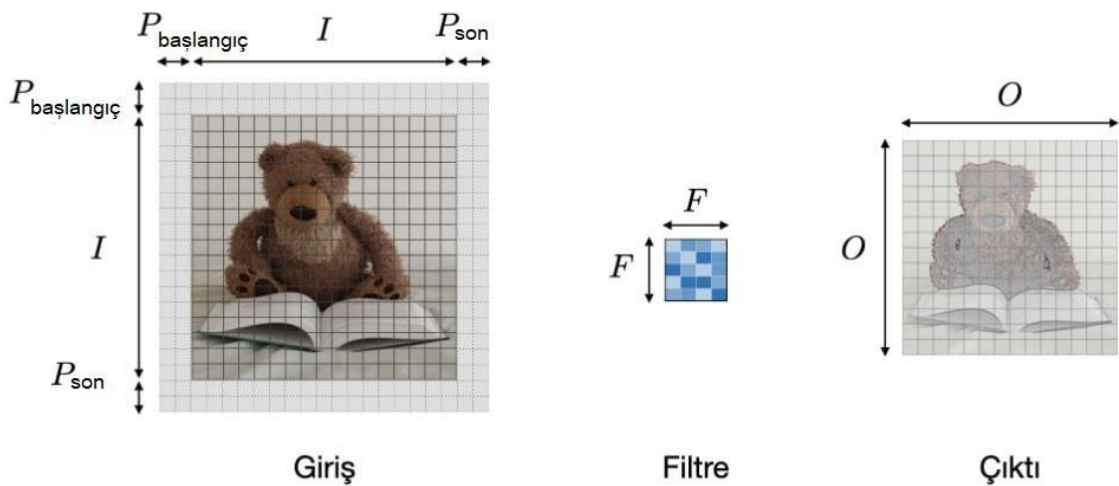
Doldurma ya da piksel ekleme, girişin sınırlarının her bir tarafına P doldurma işlemini belirtir. Genellikle doldurulacak değer olarak 0 belirlense de bu değer manuel olarak da belirlenebilir. Şekil 3.9’da detaylandırılan üç durumdan biri ile otomatik olarak ayarlanabilir.

Mod	Geçerli	Aynı	Tüm
Değer	$P = 0$	$P_{\text{start}} = \left\lfloor \frac{S \lceil \frac{I}{S} \rceil - I + F - S}{2} \right\rfloor$ $P_{\text{end}} = \left\lceil \frac{S \lceil \frac{I}{S} \rceil - I + F - S}{2} \right\rceil$	$P_{\text{start}} \in [0, F - 1]$ $P_{\text{end}} = F - 1$
Görsel Açıklama			
Amaç	- Ekleme/doldurma yok - Boyutlar uyuşmuyorsa son evrişimi düşürür	- Öznitelik harita büyüklüğüne sahip ekleme/doldurma $\left\lceil \frac{I}{S} \right\rceil$ - Çıktı boyutu matematiksel olarak uygundur 'Yarım' ekleme olarak da bilinir	- Son konvolüsyonların giriş sınırlarına uygulandığı maksimum ekleme - Filtre girişi uçtan uca 'görür'

Şekil 3.9. Doldurma işlemi için kullanılan 3 durumun görseli (Amidi ve Amidi 2018)

Evrişim sonucu oluşacak haritanın boyutu denklem 3.3. ile hesaplanabilir. Bu denklemde girdinin hacim büyüklüğünü 'I', filtresinin uzunluğunu 'F', doldurma miktarını 'P', adım aralığını 'S' daha sonra da bu boyut boyunca öznitelik haritasının çıkış büyüklüğünü 'O' belirtilir. Denklem 3.3'e ait görsel açıklama Şekil 3.10'da sunulmuştur (Amidi ve Amidi 2018).

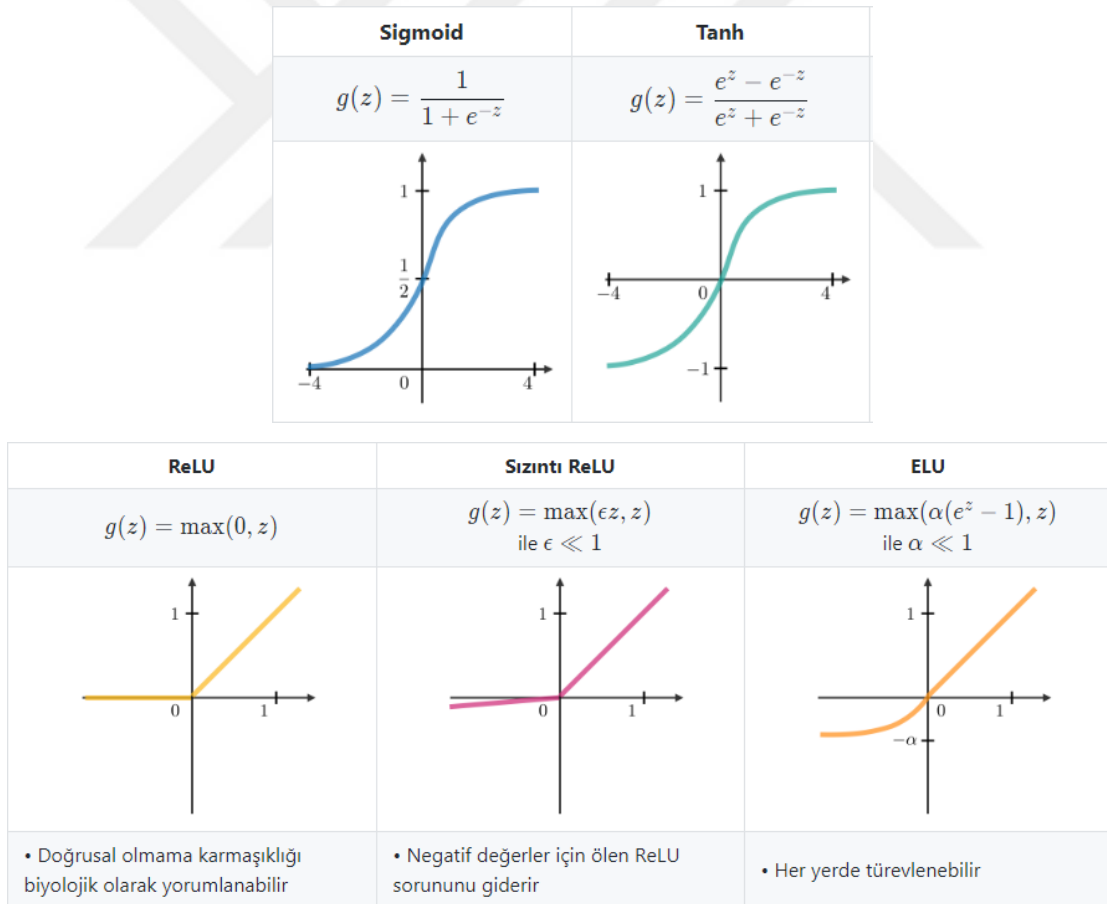
$$O = (I + F + P_{\text{başlangıç}} + P_{\text{son}}) / S + 1 \quad (3.3)$$



Şekil 3.10. Evrişim sonucu oluşacak haritanın boyutu (Amidi ve Amidi 2018)

3.2.3. Aktivasyon katmanı

Bu katmanda ağırlıklar ve girdiler çarpma işlemi ve toplama işleminden geçtikten sonra bir önceki katmanın çıkışını ve sonraki katmanın girişini oluşturur. Çok katmanlı YSA’larda da bulunan aktivasyon katmanında ESA’larda daha çok ReLU aktivasyon fonksiyonu kullanıldığı için ReLU katmanı adı da verilir (LeCun ve ark 2015). Doğrusal olmayan gerçek dünya problemlerine çözüm yeteneği kazandırması için ESA’larda doğrusal olmayan aktivasyon fonksiyonları tercih edilir. Geri yayılım işlemi ağırlıkların hataya göre türevi alınarak yapıldığından, bu işlemin yapılarak eğitimin gerçekleştirilmesi için kullanılacak fonksiyonların türevlenebilir olması gerekmektedir. Şekil 3.11’de en çok kullanılan aktivasyon fonksiyonlarından olan Sigmoid, Hiperbolik Tanjant (Tanh), ReLU, Sızıntı ReLU (Leaky ReLU) ve Elu aktivasyon fonksiyonları verilmiştir (Amidi ve Amidi 2018).



Şekil 3.11. En çok kullanılan aktivasyon fonksiyonları (Amidi ve Amidi 2018)

Bir ESA’da RGB formatında giriş görüntüsüne evrişim katmanında 3x3x3 boyutunda filtre uygulanmış ve çıktı olarak elde edilen öznetelik haritası ReLU

aktivasyonu fonksiyonuna giriş olarak uygulandıktan sonra oluşan son çıktının görüntüsü de Şekil 3.12’de verilmiştir (Özkan ve Ülker 2017).



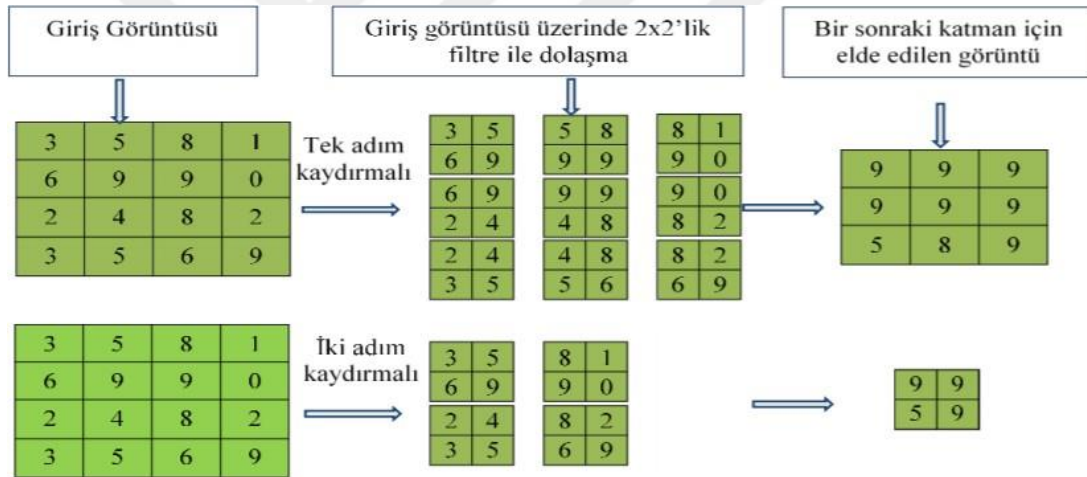
Şekil 3.12. Bir ESA modelinde ReLU katmanına ait giriş ve çıkış görüntüleri (Özkan ve Ülker 2017)

3.2.4. Ortaklama katmanı

Havuzlama katmanı olarak da bilinen bu katman tipik olarak bir miktar uzamsal değişkenlik gösteren ve genellikle aktivasyon katmanından sonra uygulanan bir örnekleme işlemidir. Maksimum ortaklama ve ortalama ortaklama, adlarından da anlaşılabilir olduğu üzere filtre üzerindeki değerlerin maksimumunun ve ortalamasının alındığı özel ortaklama türleridir. Bu işlem verideki derinlik boyutunu etkilemez. Temel amacı, sonraki katman için giriş boyutunu (Genişlik x Yükseklik) azaltmaktır. Bu sayede ESA için işlem maliyeti azalır. Diğer bir faydası ise ağıın ezberleme yapmasını önler ya da geciktirir. Fakat teorik olarak veri kaybına neden olduğu da unutulmaması gereken bir faktördür. Girdi olarak gelen matris üzerinde belirlenen boyutta ve belirlenen adımla filtreler gezdirilir. Ortaklama işleminin türüne göre filtreden bir değer çıkar. Maksimum ortaklama işleminde en büyük değer seçilirken, ortalama ortaklama işleminde filtrenin üzerinde bulunduğu değerlerin ortalaması alınır. Bu işlemler Şekil 3.13 ve Şekil 3.14’de gösterilmiştir. En çok maksimum ortaklama işlemi, (2x2 filtre boyutuyla ve 2 adım büyüklüğüyle) uygulandığı görülür (LeCun ve ark 1998, Krizhevsky ve ark 2012, Simonyan ve Zisserman 2014, LeCun ve ark 2015, Özkan ve Ülker 2017, Amidi ve Amidi 2018).

Tip	Maksimum ortaklama	Ortalama ortaklama
Amaç	Her ortaklama işlemi, geçerli matrisin maksimum değerini seçer	Her ortaklama işlemi, geçerli matrisin değerlerinin ortalaması alır
Görsel Açıklama		
Açıklama	- Algılanan özellikleri korur - En çok kullanılan	- Boyut azaltarak örneklenmiş özellik haritası - LeNet'te kullanılmış

Şekil 3.13. 6x6'lık giriş görüntüsüne 2x2 filtre ile 2 adım kaymalı maksimum ortaklama işleminin uygulanması (Amidi ve Amidi 2018)



Şekil 3.14. 5x5'lik giriş görüntüsüne 2x2 filtre ile bir ve iki adım kaymalı maksimum ortaklama işleminin uygulanması (Özkan ve Ülker 2017)

Ortaklama işleminden sonra oluşan görüntünün boyutu Denklem 3.4'e göre hesaplanır (Özkan ve Ülker 2017).

$$\text{Üretilen Görüntünün Boyutu} = G_2 \times Y_2 \times D_2 \quad (3.4)$$

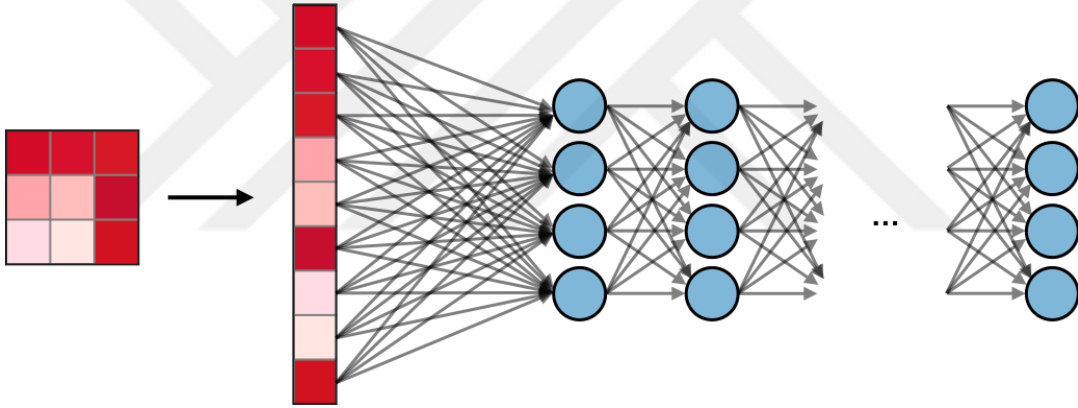
$$G_2 = (G_1 - F) / A + 1 \quad (3.5)$$

$$Y_2 = (H_1 - F) / A + 1 \quad (3.6)$$

$$D_2 = D_1 \quad (3.7)$$

3.2.5. Tam bağlantı katmanı

ESA mimarisindeki son katmandır. Bu katmanın en sonundaki sınıflandırıcı nöronların bulunduğu katmana ayrıca sınıflandırıcı katmanı da denilmektedir. Evrişim katmanından transfer katmanına, transfer katmanından ortaklama katmanına ve ortaklama katmanından da tam bağlantı katmanına ulaşan öznitelik haritaları bu katmanda işlenerek ESA'nın çıktısını oluşturur. Temel olarak çok katmanlı ağ yapılarının aynısıdır. Ortaklama katmanı çıkışında oluşan matris yapıları düzleştirilerek bu ağ katmanının nöronlarına girdi olarak verilir. Bazı ESA mimarilerinde bu katman kullanılmadan doğrudan son evrişim katmanının çıktısı da ESA'nın çıktısı olabilir. Evrişim işlemi de özünde YSA mimarisi olduğu için bu katman kullanılmayarak işlem maliyeti düşürülebilir. Şekil 3.15'de görüldüğü gibi sınıf skorları gibi hedefleri optimize etmede kullanılabilir (LeCun ve ark 1998, Amidi ve Amidi 2018).

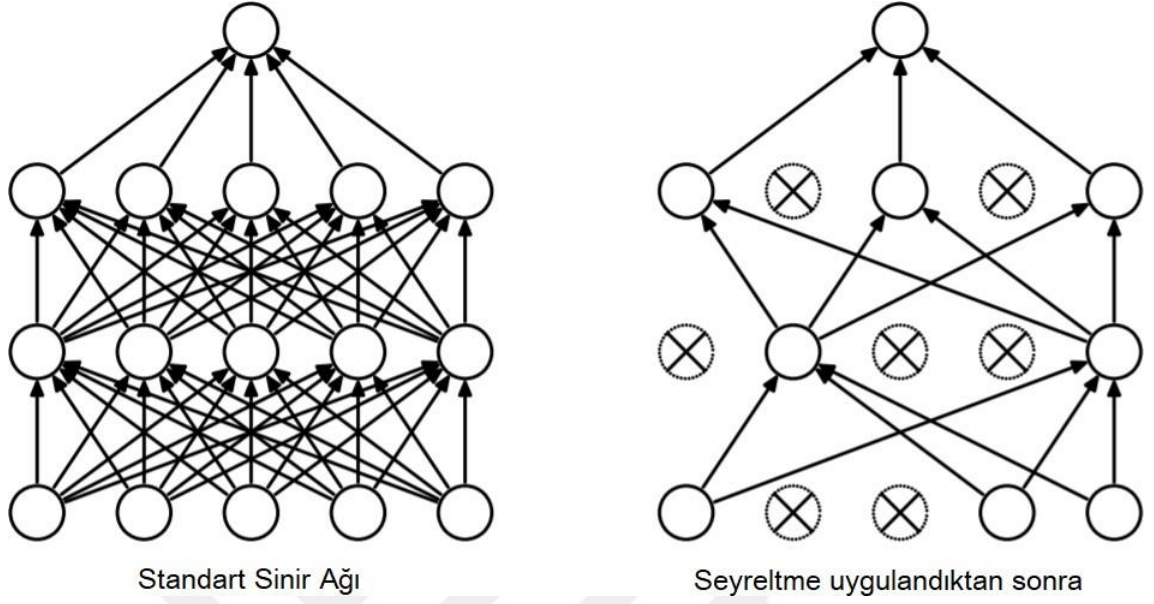


Şekil 3.15. Evrişim katmanından tam bağlantı katmanına geçiş (Amidi ve Amidi 2018)

3.2.6. Seyreltme yöntemi

Çok sayıda parametreye sahip DSA'lar çok güçlü makine öğrenme yöntemleridir. Bununla birlikte özellikle ağ yapıları büyüdükçe ve derinleştikçe ağın ezber yapması ciddi bir sorundur. Bu durum, ağın eğitim veri setinde yüksek performanslar verirken test veri setinde düşük performanslar vermesine sebebiyet verir. Seyreltme (Dropout), bu sorunun çözümü için geliştirilmiş basit ama etkili bir yöntemdir. Ana fikir ağda bulunan nöronların belirli kısmını rastgele olarak devre dışı bırakmaktır. Bu yöntem önemli ölçüde ağın ezber yapmasını azaltır. Seyreltme yöntemi ile gözetimli öğrenme üzerindeki sinir ağlarının performansını artırdığını gösteren görme, konuşma tanıma, belge sınıflandırma ve sayısal biyolojideki görevler gibi birçok veri setinde önemli sonuçlar elde edilmiştir

(Srivastava ve ark 2014). Genellikle Şekil 3.16’da olduğu gibi tam bağlantı katmanında kullanılır. Bu yöntem evrişim katmanında da kullanılabilir.



Şekil 3.16. Standart ağ modeli ve seyreltme yöntemi kullanılan model (Srivastava ve ark 2014)

3.2.7. Yığın normalleştirme yöntemi

2015 yılında ortaya atılan bu yöntem küme normalleştirilmesi olarak da bilinmektedir. DSA katmanları arasındaki değerlerin normalleştirme işlemiyle değiştirilmesi ile uygulanır. Birçok ESA modeline uygulandığında başarımlarını artırabildiği gözlemlenmiştir. Genellikle aktivasyon fonksiyonundan önce kullanılır. Seyreltme yöntemi uygulanmış ağlara da uygulanabilir. Çıktı değeri Denklem 3.8. ile hesaplanır (Ioffe ve Szegedy 2015).

$$\hat{x}_t = \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + g}} \quad (3.8.)$$

$$\sigma_\beta^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x_i - \mu_\beta)^2 \quad (3.9.)$$

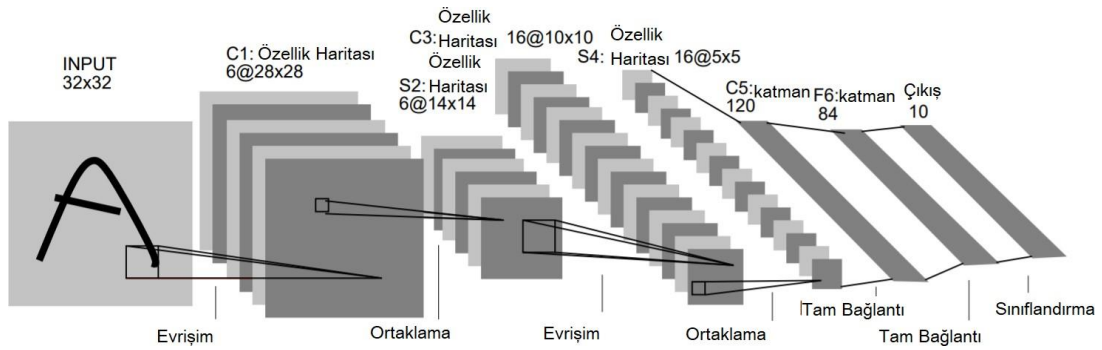
$$\mu_\beta \leftarrow \frac{1}{m} \sum_{i=1}^m x_i \quad (3.10.)$$

3.3. Evrişimsel Ağ Modelleri

1998 yılında literatürde bilinen ilk ESA modeli olan LeNet-5 ağ modeli duyurulmuştur (LeCun ve ark 1998). 2012 yılında AlexNet ESA modelinin ImageNet yarışmasında gösterdiği üstün başarıdan sonra birçok ağ modeli geliştirilmiştir ve geliştirilmektedir (Krizhevsky ve ark 2012). Bu başlık altında tezde kullanılacak olan LeNet-5, VGG-11 ve VGG-16 ESA modelleri incelenecektir.

3.3.1. LeNet-5 ağ modeli

Lecun ve arkadaşları tarafından 1998 yılında MNIST veri setini sınıflandırmak için tasarlanan ilk başarılı ESA modeli LeNet-5 modelidir. Bu modelde ReLU aktivasyon fonksiyonu kullanılmamıştır. Onun yerine sigmoid ve hiperbolik tanjant fonksiyonları kullanılmıştır. Ortaklama katmanında da ortalama ortaklama yöntemi seçilmiştir. Şekil 3.17’de görüldüğü gibi ağ 32x32 boyutlu giriş katmanına, 2 adet evrişim katmanına, 2 adet ortaklama katmanına, 2 adet tam bağlantı katmanına ve 10 nöronlu çıkışa sahiptir. Evrişim katmanında 5x5 boyutlu filtreler 1 adım kaydırılarak evrişim işlemi yapılmıştır. Ortaklama katmanında 2x2 filtreler 2 adım kaydırılarak ortalama ortaklama yapılmıştır. Ağ, 10 ve daha az sınıflı veri setlerini sınıflandırmak üzere tasarlanmıştır (LeCun ve ark 1998).

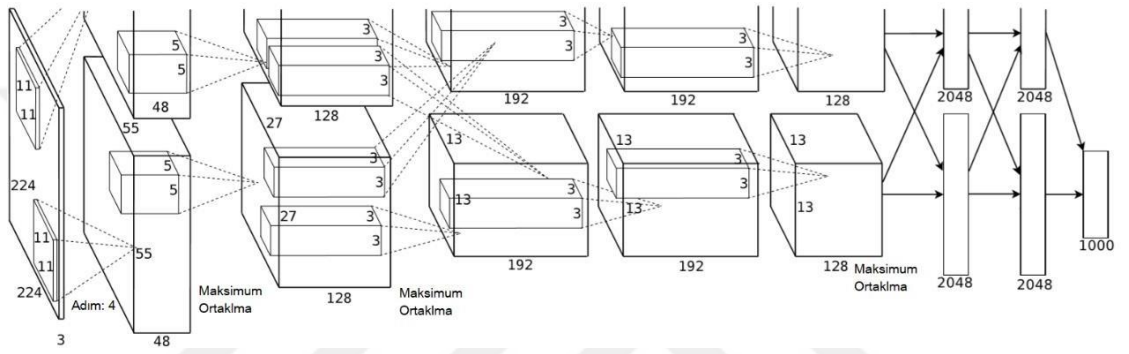


Şekil 3.17. LeNet-5 ESA modeli (LeCun ve ark 1998)

Bu ağ toplamda yaklaşık 60K parametreye sahiptir. Bu değer günümüz ESA'larına göre oldukça azdır.

3.3.2. AlexNet ağ modeli

AlexNet ağ modeli Alex Krizhevsky, Ilya Sutskever ve Geoffrey Hinton tarafından geliştirilmiştir. Birbirini takip eden evrişim ve havuzlama katmanları bulunmasından dolayı temel olarak LeNet-5 modeline çok benzemektedir. Bu modelde ReLU aktivasyon fonksiyonu kullanılmıştır. Ortaklama katmanlarında maksimum ortaklama kullanılmaktadır. Daha büyük ve daha derin olan bu ağ modeli Şekil 3.18.'de görüldüğü üzere paralel çift GPU üzerinde çalıştırılmıştır. Günümüzde tek GPU üzerinde de rahatlıkla çalıştırılabilmektedir (Krizhevsky ve ark 2012).



Şekil 3.18. AlexNet Ağ Modeli (Krizhevsky ve ark 2012)

Çizelge 3.1'de görüldüğü üzere 3x224x224 boyutlu giriş katmanına, 2 adet 4096 nörondan oluşan ve 1 adet 1000 nörondan oluşan toplam 3 adet tam bağlantı katmanına sahiptir. Ağ, 1000 ve daha az sınıflı veri setlerini sınıflandırmak üzere tasarlanmıştır. İlk evrişim katmanında 11x11 boyutlu filtreler 4 adım kaydırılarak evrişim işlemi yapılmıştır. Adım ve filtre boyutları yüksek tutularak çıkış verisinin boyutu düşürülmüştür. Bu sayede işlem maliyeti azaltılmıştır. Sonrasındaki ilk evrişim katmanında 5x5 filtreler ve takip eden diğer evrişim katmanlarında 3x3 boyutlu filtreler 1 adım kaydırılarak ve evrişim işlemi yapılmaktadır. Ortaklama katmanında 3x3 filtreler 2 adım kaydırılarak maksimum ortaklama yapılmıştır. Toplamda 3 adet ortaklama katmanına sahiptir. Çizelge 3.1'de katmanlarda bulunan parametre sayıları verilmiştir. AlexNet ağ modeli yaklaşık 62 milyon parametreye sahiptir (Krizhevsky ve ark 2012).

Çizelge 3.1. AlexNet ağ modellerinin yapısı ve parametrelerinin sayıları

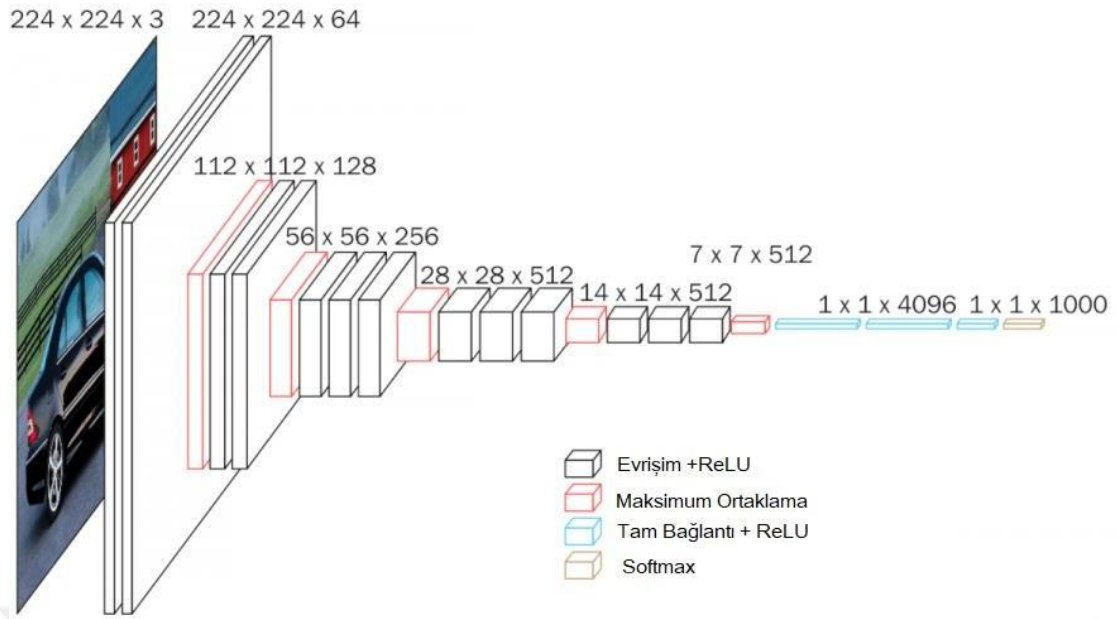
AlexNet	
Katmanlar	Parametre Sayıları
224x224 RGB görüntü	
Evrişim1 1-96	34944
Maksimum ortaklama	
Evrişim5-256	614656
Maksimum ortaklama	
Evrişim3-384	885120
Evrişim3-384	1327488
Evrişim3-256	884992
Maksimum ortaklama	
Tam Bağlantı- 4096	37752832
Tam Bağlantı- 4096	16781312
Tam Bağlantı- 1000	4097000
Softmax	Toplam
	61728744

3.3.3. VGG ağ modelleri

ESA modellerinde katman sayısının başarıya etkisini görmek amacıyla Simonyan ve Zisserman tarafından yapılmış olan çalışmada VGG-11, VGG-13, VGG-16 ve VGG-19 modelleri sunulmuştur. Çizelge 3.2’de görüldüğü üzere ağların hepsi 3x224x224 boyutlu giriş katmanına, 2 adet 4096 nöronlu oluşan ve 1 adet 1000 nöronlu oluşan toplam 3 adet tam bağlantı katmanına sahiptir. Ağlar, 1000 ve daha az sınıflı veri setlerini sınıflandırmak üzere tasarlanmıştır. Evrişim katmanlarında 3x3 boyutlu filtreler 1 adım kaydırılarak ve evrişim işlemi yapılmaktadır. Doldurma işlemi 1x1 olarak belirlenmiştir. Ortaklama katmanında 2x2 filtreler 2 adım kaydırılarak maksimum ortaklama yapılmıştır. Toplamda 5 adet ortaklama katmanına sahiptir. Şekil 3.19’da VGG-16 ağına ait görsel verilmiştir (Simonyan ve Zisserman 2014) .

Çizelge 3.2. VGG ağ modellerinin yapısı

VGG-11	VGG-13	VGG-16	VGG-19
11 adet ağırlıklı katman	13 adet ağırlıklı katman	16 adet ağırlıklı katman	19 adet ağırlıklı katman
224x224 RGB görüntü			
Evrişim3-64	Evrişim3-64	Evrişim3-64	Evrişim3-64
	Evrişim3-64	Evrişim3-64	Evrişim3-64
Maksimum ortaklama			
Evrişim3-128	Evrişim3-128	Evrişim3-128	Evrişim3-128
	Evrişim3-128	Evrişim3-128	Evrişim3-128
Maksimum ortaklama			
Evrişim3-256	Evrişim3-256	Evrişim3-256	Evrişim3-256
Evrişim3-256	Evrişim3-256	Evrişim3-256	Evrişim3-256
		Evrişim3-256	Evrişim3-256
			Evrişim3-256
Maksimum ortaklama			
Evrişim3-512	Evrişim3-512	Evrişim3-512	Evrişim3-512
Evrişim3-512	Evrişim3-512	Evrişim3-512	Evrişim3-512
		Evrişim3-512	Evrişim3-512
			Evrişim3-512
Maksimum ortaklama			
Evrişim3-512	Evrişim3-512	Evrişim3-512	Evrişim3-512
Evrişim3-512	Evrişim3-512	Evrişim3-512	Evrişim3-512
		Evrişim3-512	Evrişim3-512
			Evrişim3-512
Maksimum ortaklama			
Tam Bağlantı- 4096			
Tam Bağlantı- 4096			
Tam Bağlantı- 1000			
Softmax			



Şekil 3.19. VGG-16 Ağ Modeli

VGG-11 ağ modelinde 8 adet, VGG-13 ağ modelinde 10 adet, VGG-16 ağ modelinde 13 adet, VGG-19 ağ modelinde 16 adet evrişim katmanı bulunmaktadır. Her bir ağa ait parametre sayısı Çizelge 3.3'te verilmiştir. VGG modelleri parametre sayıları bakımından en büyük ağ modellerindendir denilebilir.

VGG-11	VGG-13	VGG-16	VGG-19
Parametre Sayıları	Parametre Sayıları	Parametre Sayıları	Parametre Sayıları
224x224 RGB görüntü			
1728	1728	1728	1728
	36864	36864	36864
Maksimum ortaklama			
73728	73728	73728	73728
	147456	147456	147456
Maksimum ortaklama			
294912	294912	294912	294912
589824	589824	589824	589824
		589824	589824
			589824
Maksimum ortaklama			
1179648	1179648	1179648	1179648
2359296	2359296	2359296	2359296
		2359296	2359296
			2359296
Maksimum ortaklama			
2359296	2359296	2359296	2359296
2359296	2359296	2359296	2359296
		2359296	2359296
			2359296
Maksimum ortaklama			
	102760448		
	16777216		
	4096000		
Toplam	Toplam	Toplam	Toplam
132851392	133035712	138344128	143652544

Bu tez çalışmasında katmanlardaki parametre sayılarının azaltmanın ve çoğaltmanın ikilileştirme yöntemlerine etkisini incelemek için VGG ağ modelleri ve AlexNet ağ modellerinin parametreleri azaltılarak yeni ağ modelleri tasarlanmıştır. Bu modeller AlexNet-k, VGG-11-k, VGG-13-k, VGG-16-k ve VGG-19-k olarak isimlendirilmiştir. Çizelge 3.4’te görüldüğü üzere ağların hepsi 3x32x32 boyutlu giriş katmanına, 2 adet 1024 nörondan oluşan ve 1 adet 100 nörondan oluşan toplam 3 adet tam bağlantı katmanına sahiptir. Ağlar, 100 ve daha az sınıflı veri setlerini sınıflandırmak üzere tasarlanmıştır. Evrişim katmanlarında 3x3 boyutlu filtreler 1 adım kaydırılarak ve

evrişim işlemi yapılmaktadır. Doldurma işlemi 1x1 olarak belirlenmiştir. Ortaklama katmanında 2x2 filtreler 2 adım kaydırılarak maksimum ortaklama yapılmıştır.

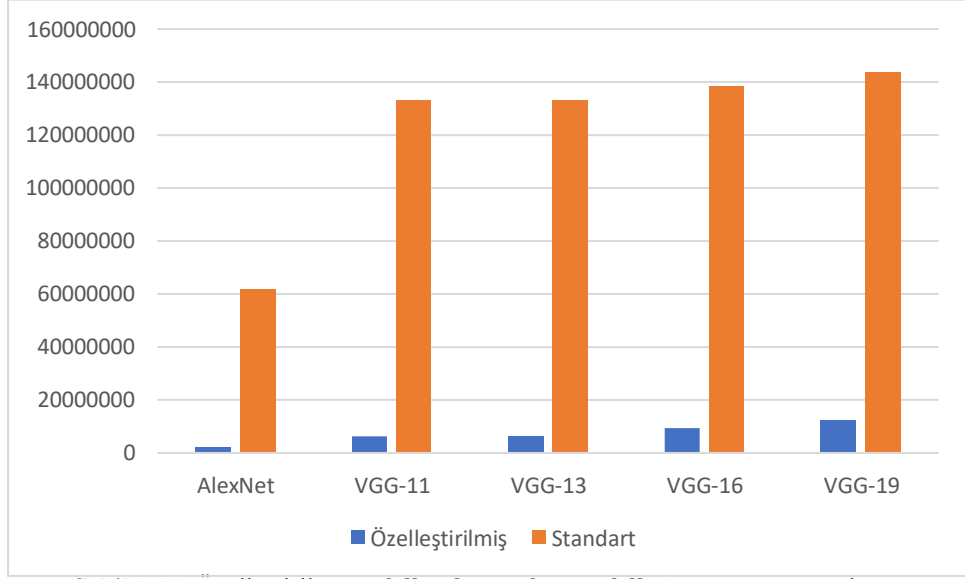
Çizelge 3.4. Özelleştirilmiş ağ modellerinin yapısı

AlexNet-k	VGG-11-k	VGG-13-k	VGG-16-k	VGG-19-k
8 adet ağırlıklı katman	11 adet ağırlıklı katman	13 adet ağırlıklı katman	16 adet ağırlıklı katman	19 adet ağırlıklı katman
224x224 RGB görüntü				
Evrişim3-48	Evrişim3-32	Evrişim3-32	Evrişim3-32	Evrişim3-32
		Evrişim3-32	Evrişim3-32	Evrişim3-32
Maksimum ortaklama				
Evrişim3-128	Evrişim3-64	Evrişim3-64	Evrişim3-64	Evrişim3-64
		Evrişim3-64	Evrişim3-64	Evrişim3-64
Maksimum ortaklama				
Evrişim3-192	Evrişim3-128	Evrişim3-128	Evrişim3-128	Evrişim3-128
Evrişim3-192	Evrişim3-128	Evrişim3-128	Evrişim3-128	Evrişim3-128
Evrişim3-128			Evrişim3-128	Evrişim3-128
				Evrişim3-128
-	Maksimum ortaklama			
-	Evrişim3-256	Evrişim3-256	Evrişim3-256	Evrişim3-256
-	Evrişim3-256	Evrişim3-256	Evrişim3-256	Evrişim3-256
-			Evrişim3-256	Evrişim3-256
-				Evrişim3-256
-	Maksimum ortaklama			
-	Evrişim3-512	Evrişim3-512	Evrişim3-512	Evrişim3-512
-	Evrişim3-512	Evrişim3-512	Evrişim3-512	Evrişim3-512
-			Evrişim3-512	Evrişim3-512
-				Evrişim3-512
Maksimum ortaklama				
Tam Bağlantı- 1024				
Tam Bağlantı- 1024				
Tam Bağlantı- 100				
Softmax				

Özelleştirilmiş ağ modellerine ait parametre sayıları Çizelge 3.5’te verilmiştir. Şekil 3.20’de özelleştirilmiş ağ modelleri ile standart yapılarına ait toplam parametre sayılarının karşılaştırması verilmiştir.

Çizelge 3.5. Özelleştirilmiş ağ modellerinin parametre sayıları

AlexNet-k	VGG-11-k	VGG-13-k	VGG-16-k	VGG-19-k
8 adet ağırlıklı katman	11 adet ağırlıklı katman	13 adet ağırlıklı katman	16 adet ağırlıklı katman	19 adet ağırlıklı katman
224x224 RGB görüntü				
1296	864	864	864	864
		9216	9216	9216
Maksimum ortaklama				
55296	18432	18432	18432	18432
		36864	36864	36864
Maksimum ortaklama				
221184	73728	73728	73728	73728
331776	147456	147456	147456	147456
221184			147456	147456
				147456
-	Maksimum ortaklama			
-	294912	294912	294912	294912
-	589824	589824	589824	589824
-			589824	589824
-				589824
-	Maksimum ortaklama			
-	1179648	1179648	1179648	1179648
-	2359296	2359296	2359296	2359296
-			2359296	2359296
-				2359296
Maksimum ortaklama				
131072		524288		
		1048576		
		10240		
Toplam	Toplam	Toplam	Toplam	Toplam
2123352	6247264	6293344	9389920	12486496



Şekil 3.20. Özelleştirilen modeller ile standart modellerin parametre sayıları

3.4. Veri Setleri

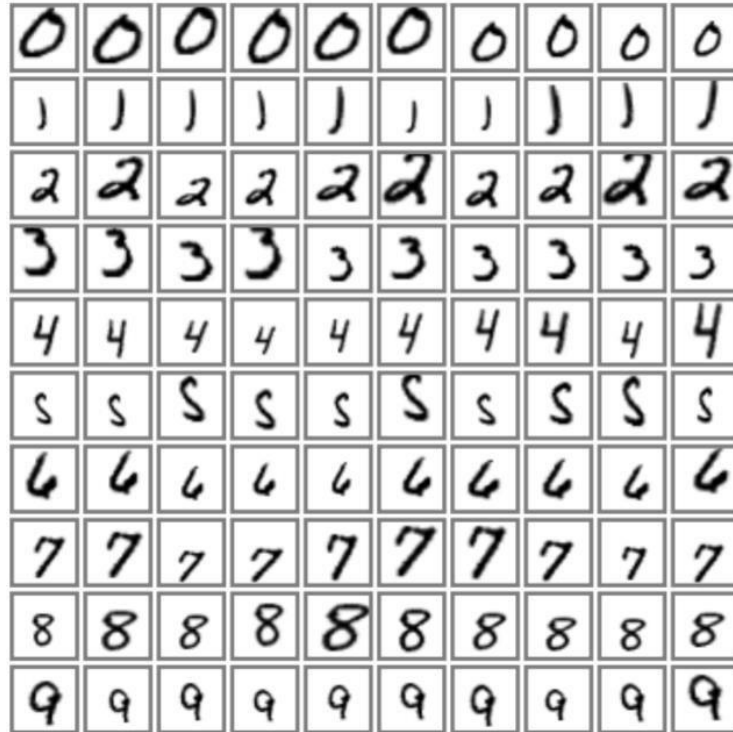
Bu çalışmada, çeşitli yerlerde toplanmış çeşitli sınıfları barındıran gri ve RGB görüntülerden oluşan birden fazla veri seti kullanılmıştır.

3.4.1. MNIST veri seti

MNIST veri seti el yazısı ile yazılmış 0'dan 9'a kadar 70.000 adet 28x28 boyutlu gri format görüntüden oluşmaktadır. Şekil 3.21'de ve Şekil 3.22'de veri setine ait 100'er adet görüntü sunulmuştur. Veri setinin bugün ki versiyonu ilk olarak 1998 yılında LeNet-5 ağının sunulduğu makale ile birlikte kullanılmıştır. Veri setinin 60.000 adet örneği eğitim kümesi olarak ayrılmıştır. Geriye kalan 10.000 örnek ise test kümesi olarak ayrılmıştır. Hem test hem de eğitim veri setinden her sınıfa ait örnek sayısı eşittir. Günümüzde tüm kullanıcıların aynı eğitim ve test kümelerine erişebildiği halka açık bir veri setidir (LeCun ve ark 1998).



Şekil 3.21. MNIST veri setine ait örnekler (LeCun ve ark 1998)



Şekil 3.22. MNIST veri seti sınıflarına ait görüntüler (LeCun ve ark 1998)

3.4.2. SVHN veri seti

SVHN veri seti, ‘Google Sokak Görüntüleri’ resimlerindeki bina numaralarından alınmış görüntülerden oluşmaktadır. Veri ön işleme ve biçimlendirme konusunda minimum gereksinimle makine öğrenimi ve nesne tanıma algoritmaları geliştirmek için kullanılan gerçek dünyaya ait bir görüntü veri kümesidir. MNIST' e benzer bir veri seti olarak görülebilir ancak daha büyük etiketli veri büyüklüğüne sahiptir. MNIST veri seti gibi 0’den 9’a kadar gri format görüntülerden değil 1’den 10’a kadar 3x32x32 boyutlu RGB format görüntülerden oluşmaktadır. Veri setinin 73.257 adet örneği eğitim kümesi olarak ayrılmıştır. Geriye kalan 26.032 örnek ise test kümesi olarak ayrılmıştır. Bu sayılar çalışmanın yapıldığı zamana aittir. Halen güncellenebilen bir veri seti olduğu için ileri tarihlerde değişiklik gösterebilir. Şekil 3.23’te veri setine ait 100 adet örnek sunulmuştur. Yukarıdan aşağıya doğru satır satır inildiğinde 1’den 10’a kadar tüm sınıflara ait 10 adet örnek bulunmaktadır. Günümüzde tüm kullanıcıların aynı eğitim ve test kümelerine erişebildiği halka açık bir veri setidir (Netzer ve ark 2011).

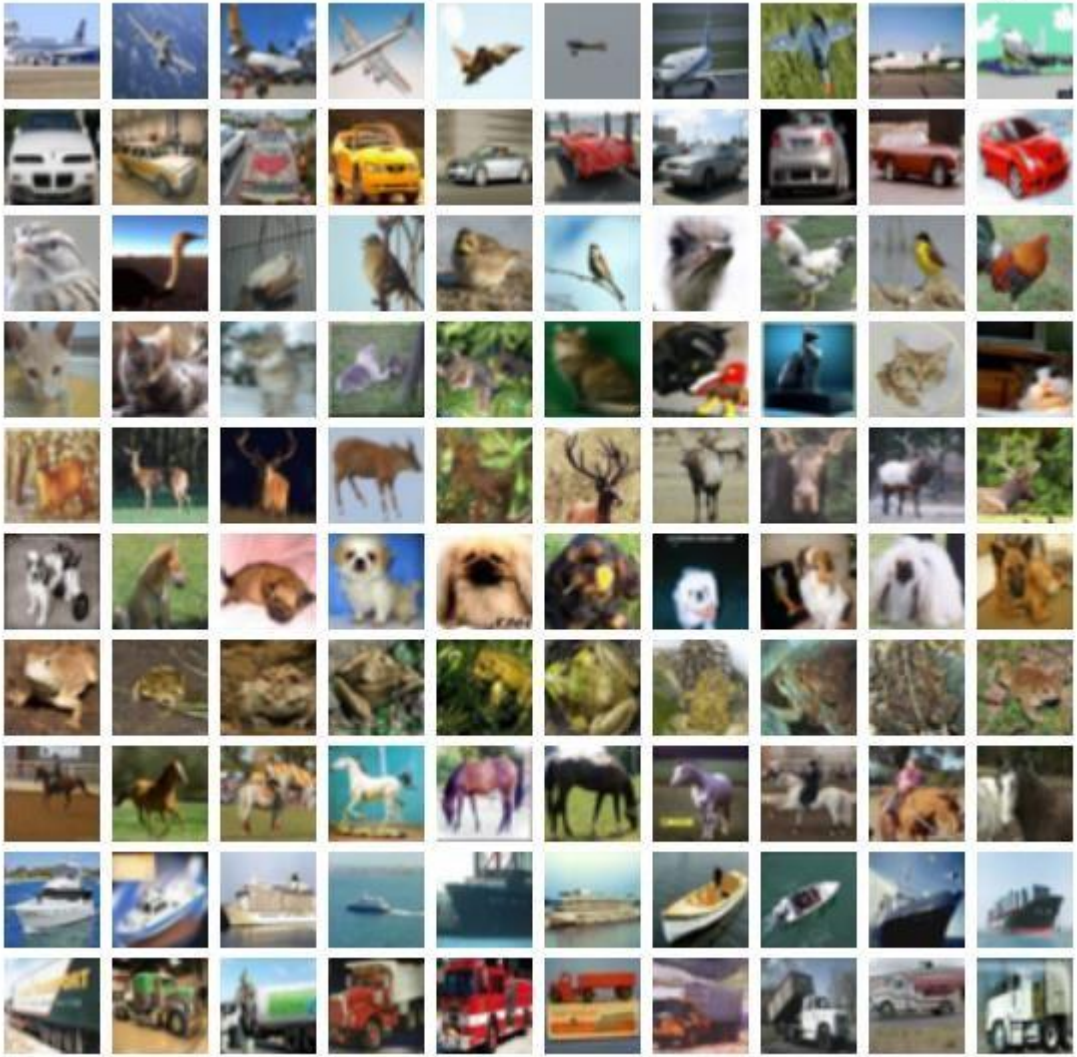


Şekil 3.23. SVHN veri seti sınıflarına ait görüntüler(Netzer ve ark 2011)

3.4.3. CIFAR-10 veri seti

AlexNet ağ modelini tasarlayan bilim adamlarından Krizhevsky ve Hinton’un 2009 yılında CIFAR-100 veri seti ile birlikte duyurduğu 3x32x32 boyutlu RGB

görüntülerden oluşan veri setidir. Veri setinde 50.000 eğitim görüntüsü ve 10.000 test görüntüsü olmak üzere 60.000 adet görüntü bulunmaktadır. Hem test hem de eğitim veri setinde her sınıfa ait örnek sayısı eşittir. Uçak, otomobil, kuş, kedi, geyik, köpek, kurbağa, at, gemi, kamyon olmak üzere 10 adet sınıfa sahiptir. Bu sınıflara ait görsel Şekil 3.24’te sunulmuştur. Her satır bir sınıfa ait örnek görüntüleri barındırır. Günümüzde tüm kullanıcıların aynı eğitim ve test kümelerine erişebildiği halka açık bir veri setidir (Krizhevsky ve Hinton 2009).



Şekil 3.24. CIFAR-10 veri seti sınıflarına ait görüntüler(Krizhevsky ve Hinton 2009)

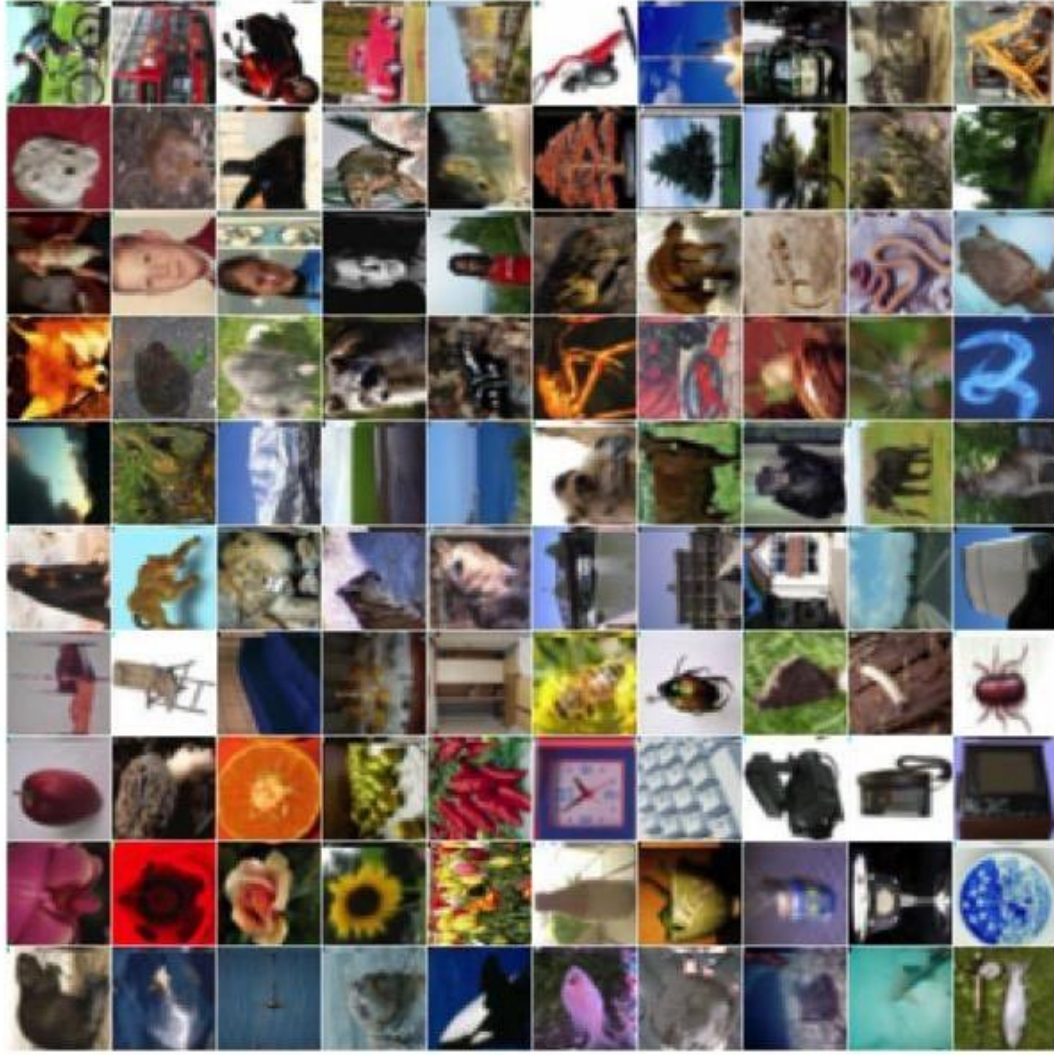
3.4.4. CIFAR-100 veri seti

AlexNet ağ modelini tasarlayan bilim adamlarından Krizhevsky ve Hinton’un 2009 yılında CIFAR-10 veri seti ile birlikte duyurduğu 3x32x32 boyutlu RGB

görüntülerden oluşan veri setidir. Veri setinde 50.000 eğitim görüntüsü ve 10.000 test görüntüsü olmak üzere 60.000 adet görüntü bulunmaktadır. Hem test hem de eğitim veri setinde her sınıfa ait örnek sayısı eşittir. Veri seti 100 adet sınıfa sahiptir. Bu sınıflar Çizelge 3.6’da verilmiştir. Bu sınıflara ait görsel Şekil 3.25’te sunulmuştur. Her bir örnek ayrı bir sınıfa ait görüntü barındırır. Günümüzde tüm kullanıcıların aynı eğitim ve test kümelerine erişebildiği halka açık bir veri setidir (Krizhevsky ve Hinton 2009).

Çizelge 3.6. CIFAR-100 veri setine ait sınıflar(Krizhevsky ve Hinton 2009)

Üst sınıf	Sınıflar
Suda yaşayan memeliler	Kunduz, yunus, su samuru, mühür, balina
Balık	Akvaryum balıkları, yassı balıklar, ışın, köpekbalığı, alabalık
Çiçekler	Orkide, haşhaş, gül, ayçiçeği, laleler
Gıda kapları	Şişeler, kaseler, kutular, bardaklar, tabaklar
Meyve ve sebzeler	Elma, mantar, portakal, armut, tatlı biber
Elektrikli ev aletleri	Saat, bilgisayar klavye, lamba, telefon, televizyon
Ev eşyası	Yatak, sandalye, kanepeler, tablo, dolap
Haşarat	Arı, böceği, kelebek, tırtıl, hamamböceği
Büyük etobur	Ayı, leopar, aslan, kaplan, kurt
Büyük insan yapımı açık şeyler	Köprü, kale, ev, yol, gökdelen
Büyük doğal açık sahneleri	Bulut, orman, dağ, düz, deniz
Büyük omnivorlar ve otodurlar	Deve, sığır, şempanze, fil, kanguru
Orta boy memeliler	Tilki, kirpi, posum, rakun, kokarca
Böcek olmayan omurgasızlar	Yengeç, ıstakoz, salyangoz, örümcek, solucan
İnsanlar	Bebek, oğlan, kız, erkek, kadın
Sürüngenler	Timsah, dinazor, kertenkele, yılan, kaplumbağa
Küçük memeliler	Fare, tavşan, sivri faresi, sincap
Ağaçlar	Akçaağaç, meşe, palmye, çam, söğüt
Araçlar 1	Bisiklet, otobüs, motosiklet, kamyonet, tren
Araçlar 2	Çim biçme makinesi, roket, tramvay, tank, traktör



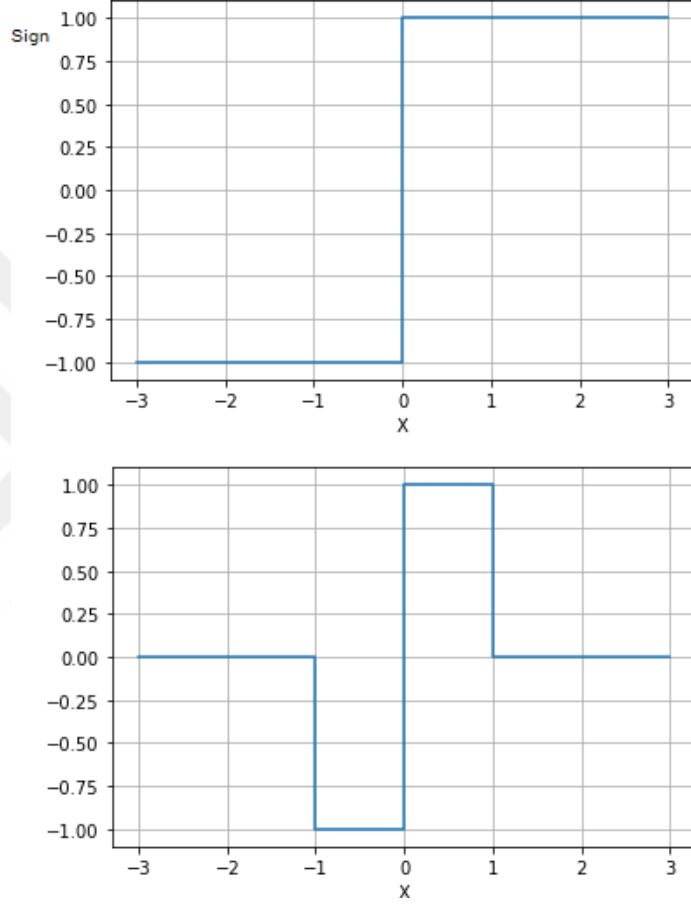
Şekil 3.25. CIFAR-100 veri seti sınıflarına ait görüntüler(Krizhevsky ve Hinton 2009)

3.5. Sinir Ağlarını İkileştirme Yöntemleri

ESA'ların yüksek bellek maliyetlerini azaltmak ve çalışma hızlarını nispeten artırmak için ağ ağırlıkları -1 ve +1 ikili değerinden birini alacak şekilde ağ eğitilebilir. Bu eğitim işlemi için ikili ağırlıklı ağ yöntemi kullanılabilir (Courbariaux ve ark 2015, Rastegari ve ark 2016). Ağırlıklar -1 ve +1 ikili değerinden biri alacak şekilde eğitilirken ayrıca Sign fonksiyonu kullanılarak giriş değerleri de ikiliye çevrilebilir. Bu sayede gerçek zamanlı uygulamalarda ikili giriş ve ikili ağırlıklara, çarpma işlemi yerine 'özel veya değil işlemi' kullanarak ağın ileri besleme hızı ciddi miktarda artırılabilir. Bunu sağlamak için XNOR-Ağ yöntemi kullanılabilir (Hubara ve ark 2016, Rastegari ve ark 2016).

Sign fonksiyonu, Denklem 3.11’de verilmiştir. Bu fonksiyon, giriş değerlerine bağlı olarak -1 ya da +1 çıkışlarından birini çıktığı olarak verir. Şekil 3.26’da fonksiyona ait giriş-çıkış grafiği ve çıkışın türevinin grafiği verilmiştir.

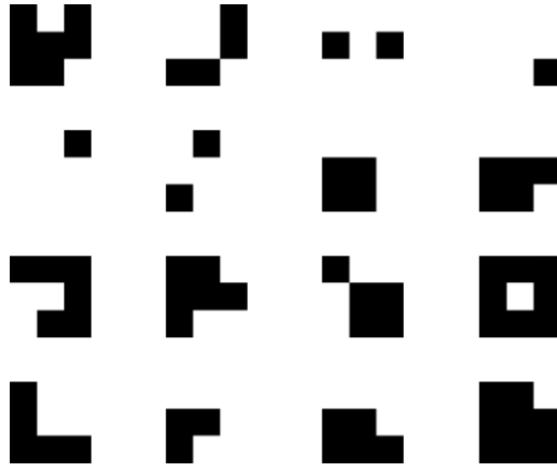
$$Sign(x) = \begin{cases} +1 & \text{eğer } x \geq 0 \\ -1 & \text{eğer } x < 0 \end{cases} \quad (3.11.)$$



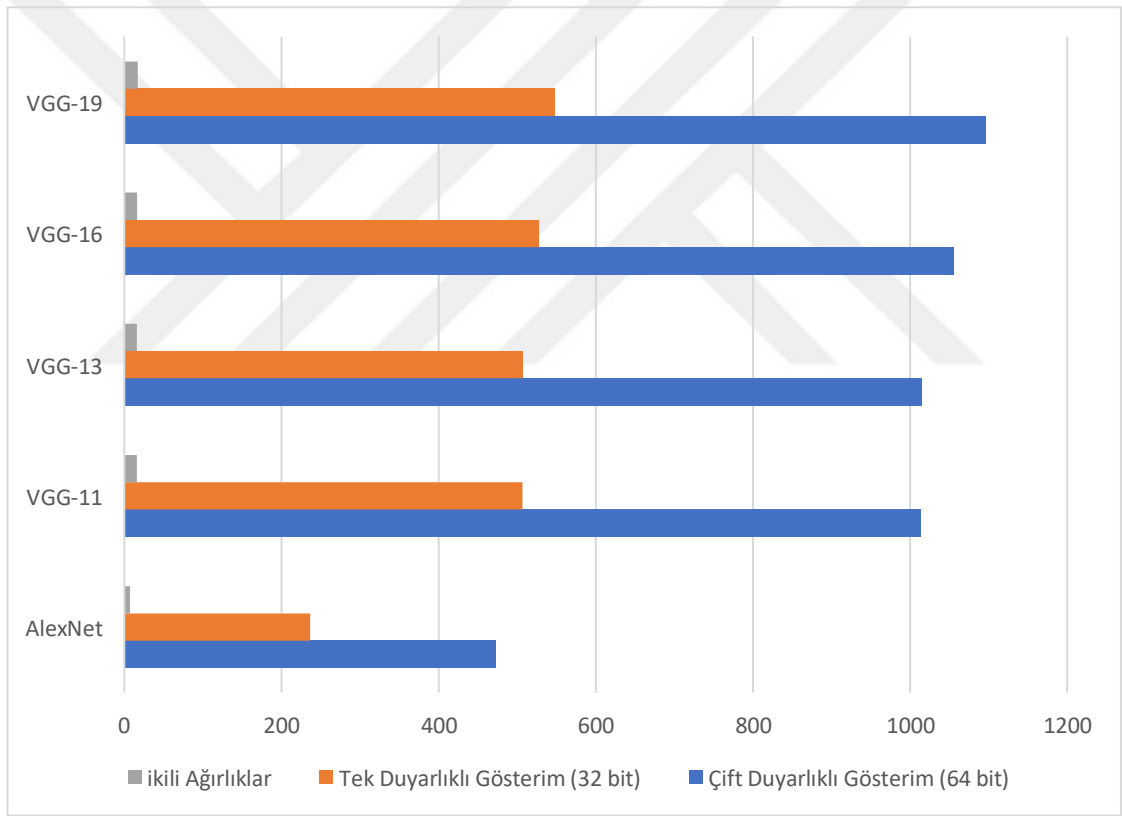
Şekil 3.26. Sign fonksiyonu ve geri yayılım için kullanılan türevi

3.5.1. İkili ağırlıklı ağ

2016 yılında Rastegari ve arkadaşları tarafından tanıtılan, ESA’nın ağırlıklarının -1 ve +1 değerlerinden birini alacak şekilde eğitildiği ağ ikilileştirme yöntemidir. Şekil 3.27’de evrişim işlemi için kullanılan ikilileştirilmiş filtreler için eğitim sonrası örnekler verilmiştir. Tek duyarlılık kayan noktalı gösterime göre yaklaşık 32 kat bellek kazancı sağlar. Gerçek zamanlı uygulamalarda yaklaşık 2 kat daha hızlı çalışır. Çeşitli ağ yapılarıyla ilgili bellek kazancına ait grafik, Şekil 3.28’de yaklaşık olarak verilmiştir (Rastegari ve ark 2016).



Şekil 3.27. İkileştirilmiş 3x3 evrişim filtrelerine ait örnekler



Şekil 3.28. İkileştirilmiş ve standart ağırlıkların MB cinsinden karşılaştırması

Çizelge 3.7’de bellek erişimi için harcanan enerji miktarları verilmiştir. Buna göre bellek tasarrufu sağlandığında aynı zamanda enerji tasarrufu da sağlanmış olur.

Çizelge 3.7. Bellek erişimlerinde harcan enerji miktarları (Horowitz 2014)

Bellek Büyüklüğü	64-bit Bellek Erişimi
8K	10pJ
32K	20pJ
1M	100pJ
DRAM	1.3-2.6nJ

İkili ağırlıklar ve XNOR işlemi ile DSA'ların doğru sonuçlar verebilmesi için Denklem 3.11 ve 3.12'de verildiği üzere bir $\alpha \in R^+$ çarpanına ihtiyaç vardır. Bu α değeri optimum ölçekleme çarpanı olarak adlandırılır. I giriş değerlerini, W ondalıklı standart ağırlıkları, $\oplus$ çarpma işlemi içermeyen evrişim işlemi ve B ikili ağırlıkları temsil eder. Denklem 3.13'de α değerinin ilgili katmandaki ondalıklı ağırlıkların mutlak ortalaması alınarak hesaplanabileceği verilmiştir. Bu denklemde n ağırlık sayısını temsil eder (Rastegari ve ark 2016).

$$I * W \approx (I \oplus B)\alpha \quad (3.12.)$$

$$B \in \{+1, -1\} \quad (3.13.)$$

$$\alpha^* = \frac{\sum |W_i|}{n} = \frac{1}{n} \|W\|_{l1} \quad (3.14.)$$

Ağın ikili olarak eğitilmesi aşağıda sunulan Algoritma-1 ile yapılır. Bu algorithmada evrişim katmanlarında bias kullanılmaz (Rastegari ve ark 2016).

Algoritma-1 İkili ağırlıklar için eğitim:

1. Rastgele W ağırlıklarını ata
2. For iter = 1 to N
 - a. Sıradaki X görüntüsünü girişe yükle
 - b. $W^B = \text{sign}(W)$
 - c. W^B ve α le ileri besleme yap
 - d. Hata fonksiyonu ile C hatasını hesaplama
 - e. $\frac{\partial C}{\partial \alpha W^B} = dW$ İkili ağırlıklar ve optimum ölçekleme çarpanı ile eğim hesapla
 - f. $W = W - dW * l_r$ Normal ağırlıklara güncelleme yap

3.5.2. XNOR-Ağ

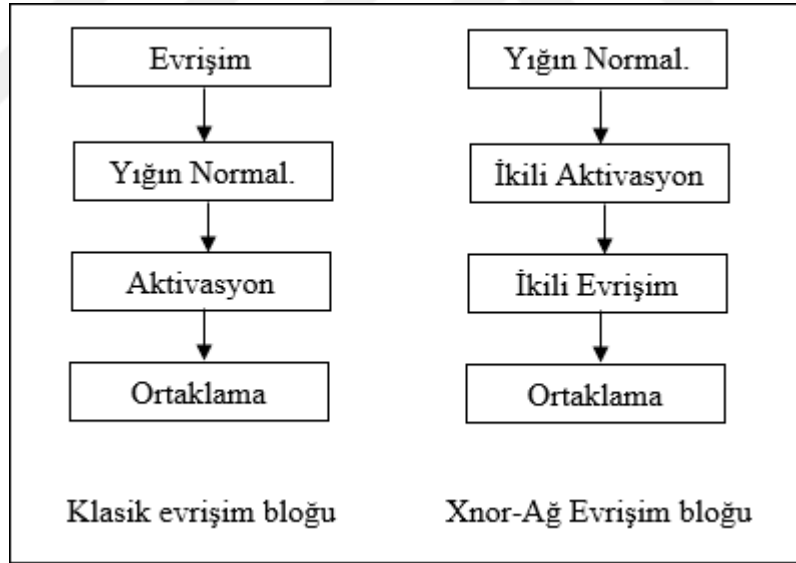
Bu yöntemde de tıpkı İAA yönteminde olduğu gibi ağırlıklar ikilileştirilir. Farklı olarak burada girişler de Sign fonksiyonu kullanılarak $\{-1, +1\}$ kümesinden bir değere

indirgenir. Denklem 3.15'te görüldüğü üzere α değerine ek olarak β değeri eklenebilir. $\odot$ “özel veya değil” işlemini temsil eder. β değeri denklem 3.16. da verildiği üzere ilgili katmandaki girişlerin mutlak ortalamasıdır (Rastegari ve ark 2016).

$$I * W \approx (I \odot B)\beta\alpha \quad (3.15.)$$

$$\beta^* = \frac{\sum |x_i|}{n} = \frac{1}{n} \|X\|_{l1} \quad (3.16.)$$

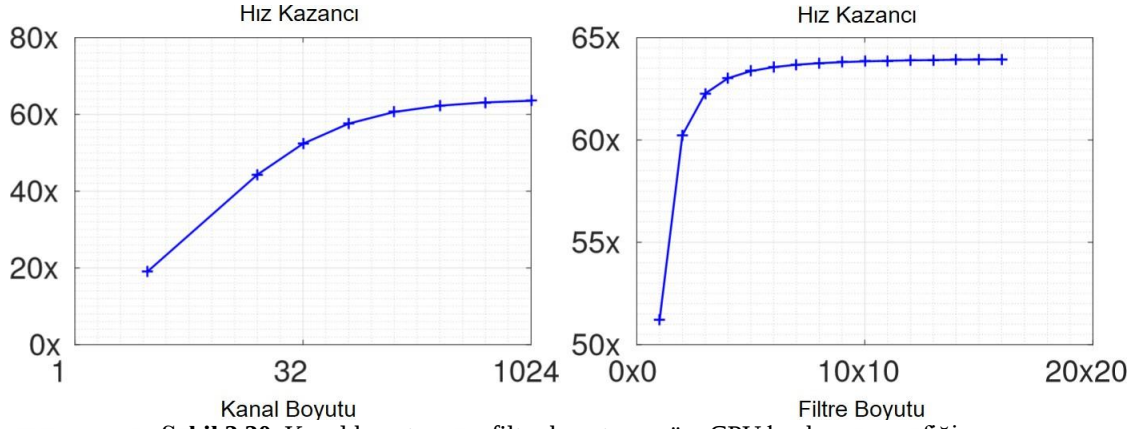
Ağın ikili olarak eğitilmesi Başlık 3.5.1 altında sunulan Algoritma-1 ile yapılır. İleri beslenme ve geri yayılım yapılırken β değeri α değeri ile birlikte kullanılabilir. Bu algorithmada evrişim katmanlarında bias kullanılmaz (Rastegari ve ark 2016). Şekil 3.29'da görüldüğü üzere klasik bir evrişim bloğu sırasıyla evrişim, yığın normalleştirme, aktivasyon ve ortaklama işlemlerinden oluşurken XNOR-Ağ bloğu sırasıyla yığın normalleştirme, İkili (Sign) aktivasyon, ikili evrişim ve ortaklama işlemlerinden oluşur (Rastegari ve ark 2016).



Şekil 3.29. Klasik evrişim bloğu ve XNOR-Ağ evrişim bloğu (Rastegari ve ark 2016)

Bu yöntem ile çarpma işlemi yerine ‘özel veya değil’ ikili operatörü kullanılarak yaklaşık 58 kat işlem kazancı sağlanmaktadır. Xilling FPGA üzerinde 32 bit ondalıklı iki sayıyı çarpma işlemi 200 dilim maliyeti oluştururken 1 bit iki adet sayının XNOR işlemi 1 dilime mâl olmaktadır (Govindu ve ark 2004, Beauchamp ve ark 2006). Bu sayede çoklu paralel işlemleri CPU üzerinden de gerçekleştirebilerek birçok taşınabilir

sistemde ESA'ların çalışabilmesine olanak sağlanmaktadır. Şekil 3.30'da kanal boyutuna ve filtre boyutuna göre CPU hız kazancının değişim grafiği verilmiştir.



Şekil 3.30. Kanal boyutuna ve filtre boyutuna göre CPU hız kazanç grafiği

Ağ parametrelerini ikilileştirmek ve çarpma işlemi yerine XNOR operatörü ile işlem yaparak enerji tasarrufu 32 kattan fazla olmaktadır (Horowitz 2014). Giriş parametrelerine karşı harcanan enerjiye ait değerler Çizelge 3.8. üzerinde verilmiştir.

Çizelge 3.8. İşlemlere ait enerji harcamaları (Horowitz 2014)

	Çarpma	Toplama
8bit Tam Sayı	0.2pJ	0.03pJ
32bit Tam Sayı	3.1pJ	0.1pJ
16bit Ondalık Sayı	1.1pJ	0.4pJ
32tbit Ondalık Sayı	3.7pJ	0.9pJ

2016 yılında İAA ve Xnor-Ağ yöntemleri ile birlikte verilen sınıflandırma doğruluğu karşılaştırması (Imagenet veri setinde AlexNet ağ modeli kullanılarak standart, ikili ağırlıklı ağ ve XNOR-Ağ yöntemleri karşılaştırılması) Çizelge 3.9'da verilmiştir (Rastegari ve ark., 2016).

Çizelge 3.9. AlexNet ağ modeli ve ImageNet veri setinde sınıflandırma doğruluğu karşılaştırması (Rastegari ve ark., 2016)

Sınıflandırma Doğruluğu		
Standart	İAA	XNOR-Ağ
%56,6	%56,8	%44,2

3.6. Veri Ön İşleme

ESA'ların en güçlü yanlarından biri de veri ön işleme işlemlerine çok fazla ihtiyaç duymamalarıdır. Fakat yine de veri normalleştirme ve veri artırma gibi ön işleme metotları da ağıın başarısını yükseltebilmektedir.

3.6.1. Veri artırma

DSA'lar uygun eğitilmesi için çok fazla sayıda veriye ihtiyaç duyan makine öğrenmesi uygulamalarıdır. Veri artırma yöntemini kullanmak genellikle ağıın performansını artırır. Sıklıkla kullanılan görüntü verisi artırma yöntemleri: dikey çevirme, açılı döndürme, rastgele kırpma, renk değişimi, gürültü ekleme, bilgi kaybı ve kontrast değişikliğidir. Bu yöntemlere ait örnek görseller Şekil 3.31'de verilmiştir (Amidi ve Amidi 2018).



Şekil 3.31. Görüntü verilerini artırma yöntemleri verilmiştir (Amidi ve Amidi 2018)

4. DENEYSEL SONUÇLAR

Bu tez çalışmasında MNIST, SVHN, CIFAR-10 ve CIFAR-100 veri setleri kullanılmıştır. LeNet-5, VGG-11, VGG-13, VGG-16, VGG-19, AlexNet, AlexNet-k, VGG-11-k, VGG-13-k, VGG-16-k ve VGG-19-k ESA modelleri kullanılmıştır. Bu modeller standart, İAA ve XNOR-Ağ yöntemleriyle kullanılarak toplamda 87 adet deney yapılmıştır.

Bu çalışmalar, Python programlama dilinde yazılmıştır. Colab üzerinde çalıştırılmıştır. Pytorch, Numpy, Matplotlib kütüphaneleri kullanılmıştır.

Bu başlık altında yapılan tüm çalışmalara ait ortak hiper parametreler:

1. Yığın normalleştirme, momentum = 0,1
2. Hata fonksiyonu = Çapraz entropi kaybı
3. Optimizasyon, SGD (momentum= 0.9, lr = 0.001)

4.1. MNIST Veri Seti Üzerinde Yapılan Deneyler

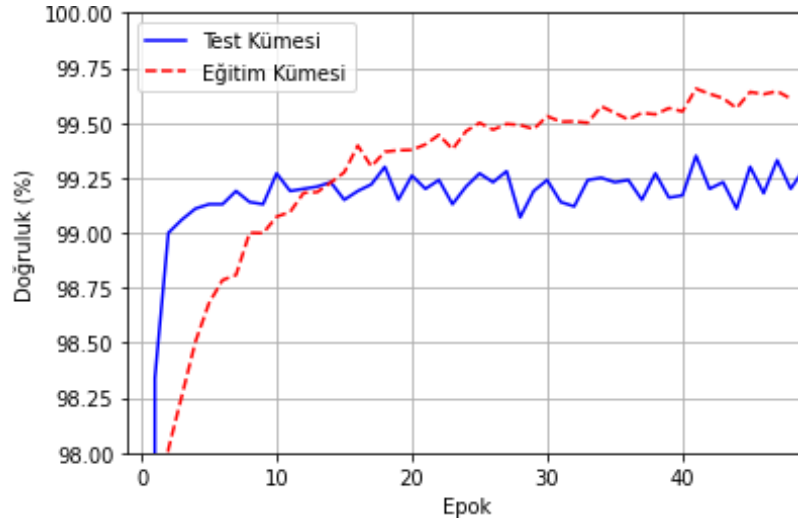
MNIST veri seti hakkında bilgiler Bölüm 3.4.1’de detaylı olarak verilmiştir. Bu veri seti üzerinde LeNet-5, VGG-11 ve VGG-19-k ESA modelleri denenmiştir.

Bu başlık altında yapılan çalışmalarda, veri setinde herhangi bir ön işleme ya da veri çoğaltma yapılmamıştır. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

4.1.1. MNIST veri seti ve LeNet-5 ağ modeli ile yapılan deneyler

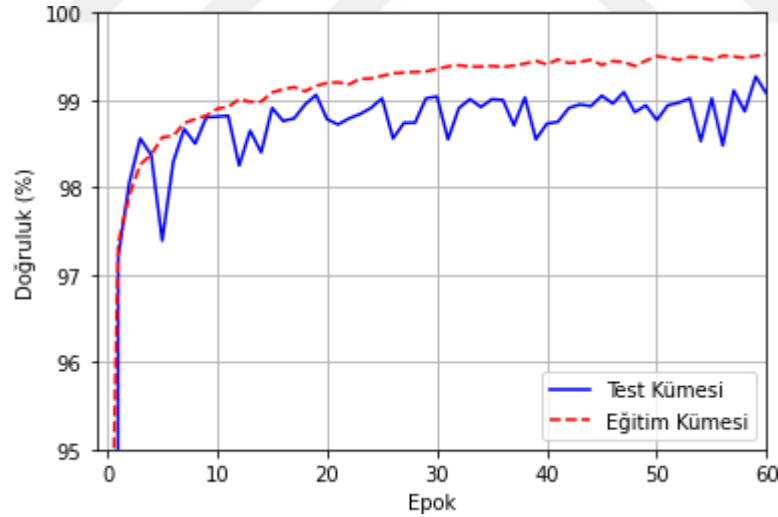
Veri seti 1x28x28 boyutlu ve ağ modelinin girişi 1x32x32 boyutlu olduğu için ilk evrişim katmanında 2x2 doldurma/sıfır ekleme işlemi yapılmıştır.

LeNet-5 ESA modeli, standart yöntemle MNIST veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.1’de verilmiştir. 50 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,655 ve test kümesinde %99,35 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



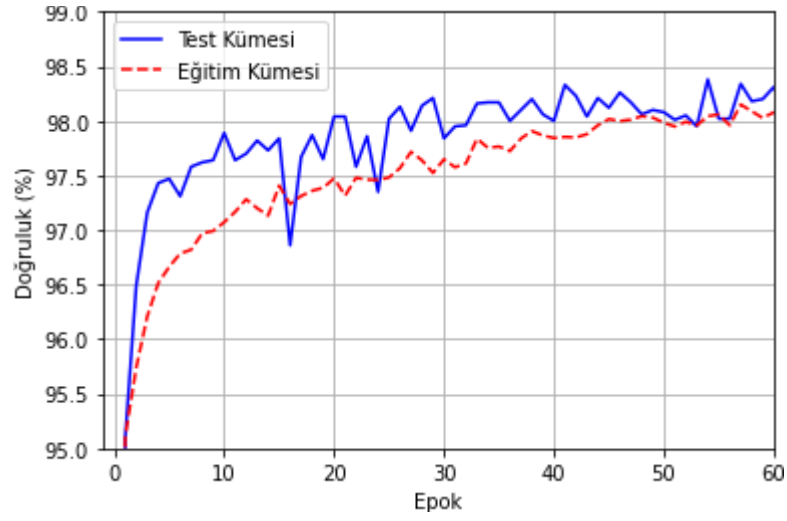
Şekil 4.1. LeNet-5 ESA modelinin standart yöntemle MNIST veri seti ile eğitimi

LeNet-5 ESA modeli, İAA yöntemiyle MNIST veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.2’de verilmiştir. 60 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,507 ve test kümesinde %99,27 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.2. LeNet-5 ESA modelinin İAA yöntemiyle MNIST veri seti kullanılarak eğitimi

LeNet-5 ESA modeli, XNOR-Ağ yöntemiyle MNIST veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.3’te verilmiştir. 60 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,15 ve test kümesinde %98,38 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.

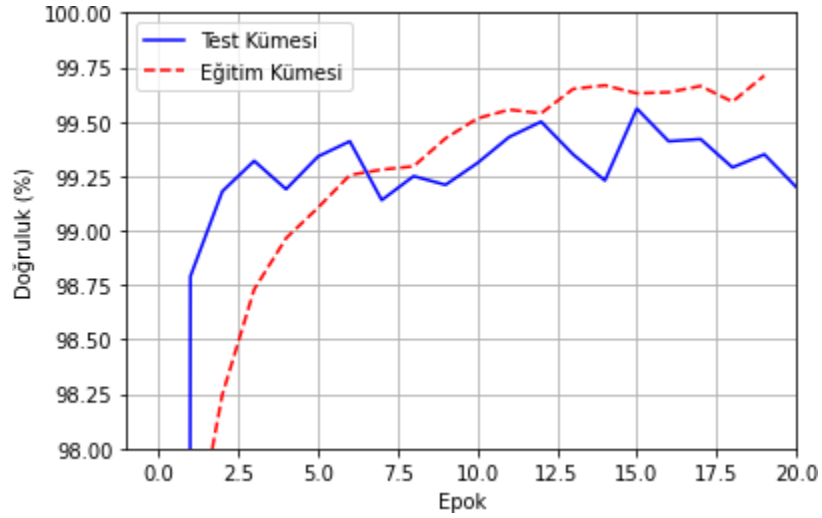


Şekil 4.3. LeNet-5 ESA modelinin XNOR-Ağ yöntemiyle MNIST veri seti kullanılarak eğitimi

4.1.2. MNIST veri seti ve VGG-11 ağ modeli ile yapılan deneyler

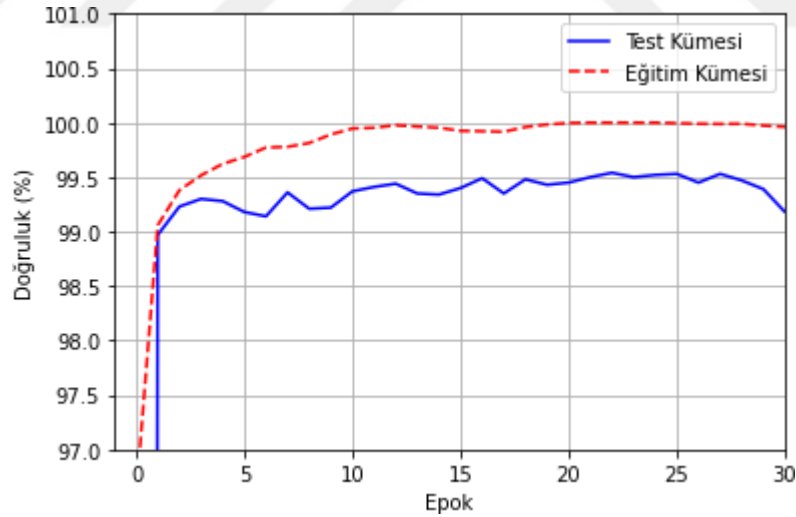
Veri seti 1x28x28 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için ilk evrişim katmanında 3x3 doldurma/sıfır ekleme yapılmıştır ve ilk katmandaki filtrelerin derinliği 3'den 1'e indirilmiştir. Tam bağlantı katmanına ulaşan öznitelik haritaları 7x7 olması gerekirken 1x1 olmaktadır. Bunun nedeni girdi olarak 224x224 boyutlu görüntü yerine 32x32 boyutlu görüntü verilmesidir. Bu yüzden son evrişim katmanı ile ilk tam bağlantı katmanı arasındaki bağlantı 25.088x4096 yerine 512x4096 olarak güncellenmiştir.

VGG-11 ESA modeli, standart yöntemle MNIST veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.4'te verilmiştir. 20 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,712 ve test kümesinde %99,56 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



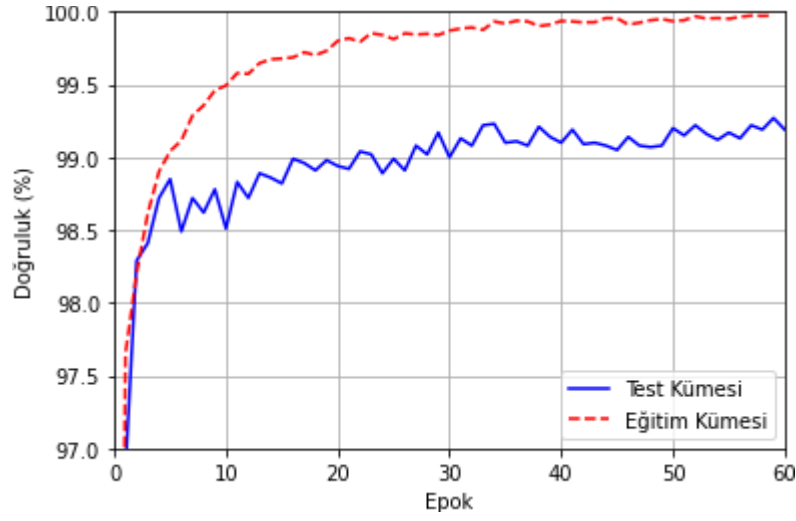
Şekil 4.4. VGG-11 ESA modelinin standart yöntemle MNIST veri seti ile eğitimi

VGG-11 ESA modeli, İAA yöntemiyle MNIST veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.5’de verilmiştir. 25 epok boyunca eğitilen ağ modeli eğitim kümesinde %100 ve test kümesinde %99,54 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.5. VGG-11 ESA modelinin İAA yöntemiyle MNIST veri seti kullanılarak eğitimi

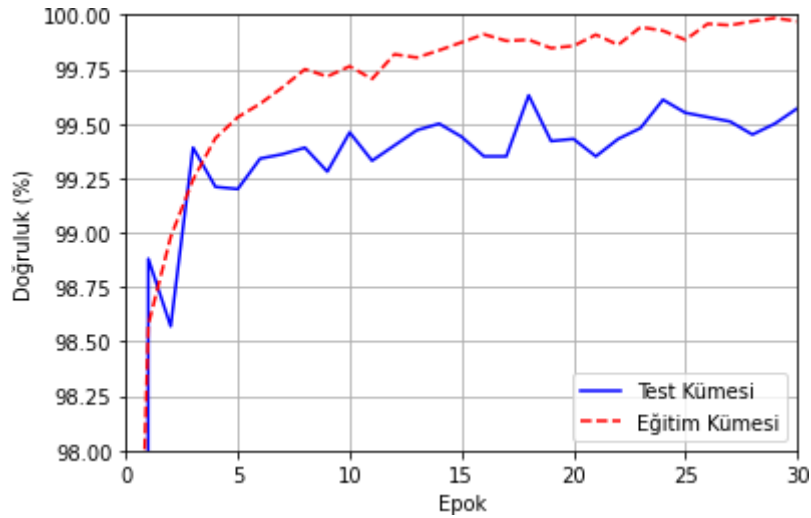
VGG-11 ESA modeli, XNOR-Ağ yöntemiyle MNIST veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.6’da verilmiştir. 35 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,973 ve test kümesinde %99,27 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.6. VGG-11 ESA modelinin XNOR-Ağ yöntemiyle MNIST veri seti kullanılarak eğitimi

4.1.3. MNIST veri seti ve VGG-19-k ağ modeli ile yapılan deneyler

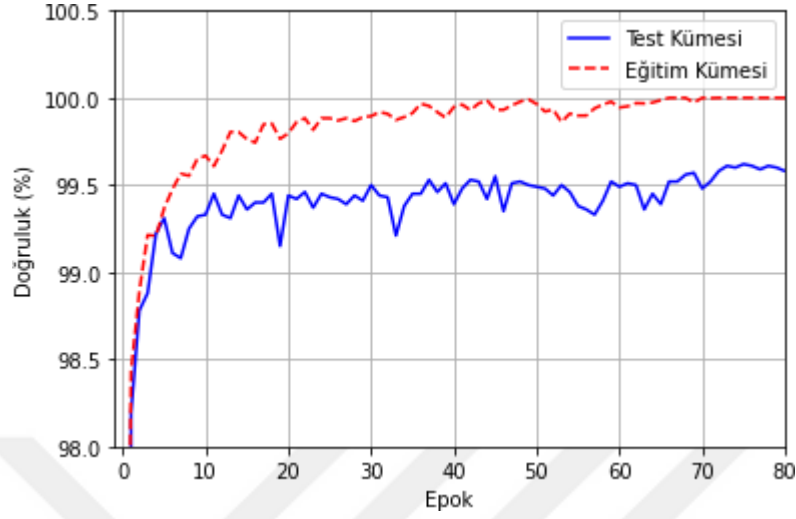
VGG-19-k ESA modeli, standart yöntemle MNIST veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.7’de verilmiştir. 30 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,985 ve test kümesinde %99,63 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



Şekil 4.7. VGG-19-k ESA modelinin standart yöntemle MNIST veri seti ile eğitimi

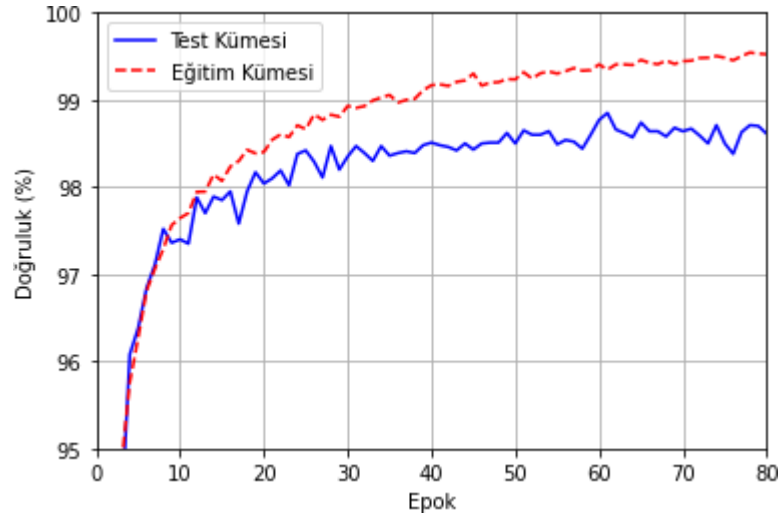
VGG-19-k ESA modeli, İAA yöntemiyle MNIST veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.8’de verilmiştir. 80 epok boyunca eğitilen ağ modeli eğitim kümesinde

%100 ve test kümesinde %99,62 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.8. VGG-19-k ESA modelinin İAA yöntemiyle MNIST veri seti kullanılarak eğitimi

VGG-19-k ESA modeli, XNOR-Ağ yöntemiyle MNIST veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.9’da verilmiştir. 80 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,627 ve test kümesinde %98,85 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



Şekil 4.9. VGG-19-k ESA modelinin XNOR-Ağ yöntemiyle MNIST veri seti kullanılarak eğitimi

4.2. SVHN Veri Seti Üzerinde Yapılan Deneyler

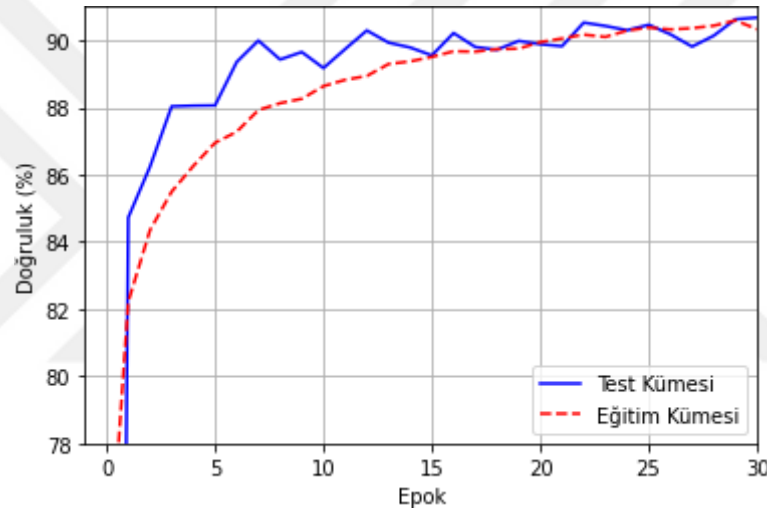
SVHN veri seti hakkında bilgiler Bölüm 3.4.2’de detaylı olarak verilmiştir. Bu veri seti üzerinde LeNet-5, VGG-11 ve V-19s ESA modelleri denenmiştir.

Bu başlık altında yapılan çalışmalarda, veri setinde herhangi bir önışleme ya da veri çoğaltma yapılmamıştır. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

4.2.1. SVHN veri seti ve LeNet-5 ağ modeli ile yapılan deneyler

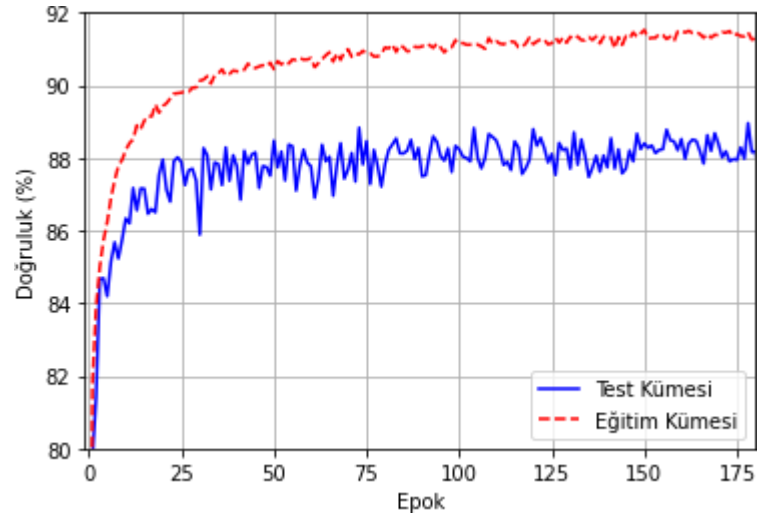
Veri seti 3x32x32 boyutlu ve ağ modelinin girişı 1x32x32 boyutlu olduđu için olduđu için ilk evriřim katmanında filtrelerin derinliđi 1'den 3'e çıkarılmıştır.

LeNet-5 ESA modeli, standart yöntemle SVHN veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değeriinin döneme (epok) göre deđiřimi Şekil 4.10'da verilmiştir. 30 epok boyunca eğitilen ağ modeli eğitim kümesinde %90,591 ve test kümesinde %90,677 sınıflandırma doğruluđuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



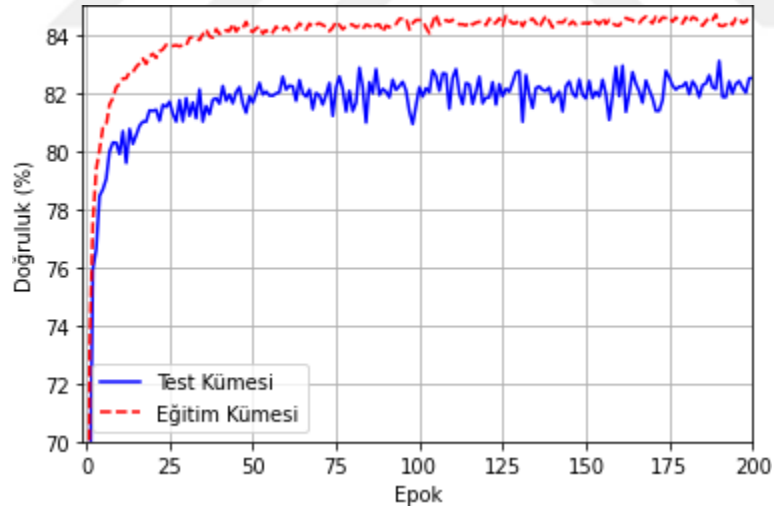
Şekil 4.10. LeNet-5 ESA modelinin standart yöntemle SVHN veri seti ile eğitimi

LeNet-5 ESA modeli, İAA yöntemiyle SVHN veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değeriinin döneme (epok) göre deđiřimi Şekil 4.11'de verilmiştir. 180 epok boyunca eğitilen ağ modeli eğitim kümesinde %91,611 ve test kümesinde %88,96 sınıflandırma doğruluđuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.11. LeNet-5 ESA modelinin İAA yöntemiyle SVHN veri seti kullanılarak eğitimi

LeNet-5 ESA modeli, XNOR-Ağ yöntemiyle SVHN veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.12’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %84,698 ve test kümesinde %83.102 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 64 olarak atanmıştır.



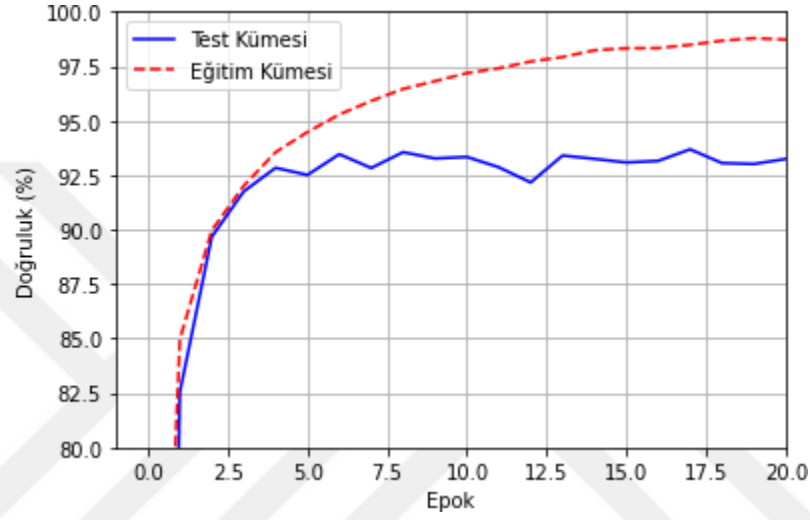
Şekil 4.12. LeNet-5 ESA modelinin XNOR-Ağ yöntemiyle SVHN veri seti kullanılarak eğitimi

4.2.2. SVHN veri seti ve VGG-11 ağ modeli ile yapılan deneyler

Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için tam bağlantı katmanına ulaşan öz nitelik haritaları 7x7 olması gerekirken 1x1 olmaktadır.

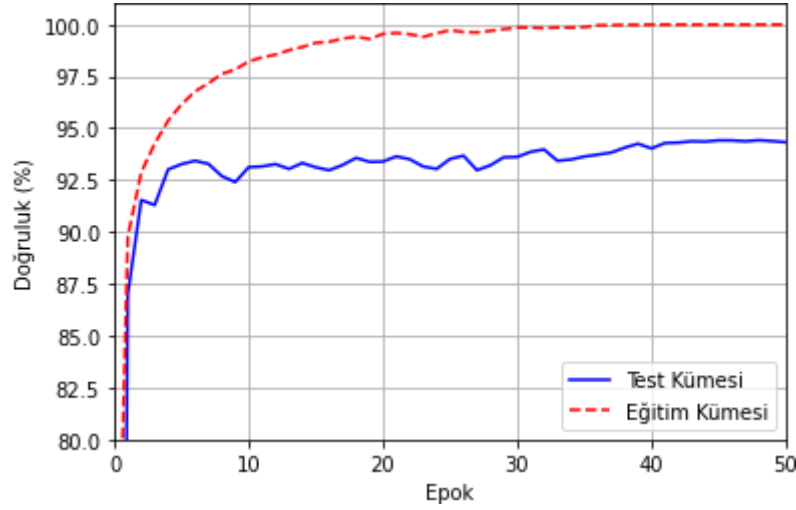
Bu yüzden son evriřim katmanı ile ilk tam baęlantı katmanı arasındaki baęlantı 25.088x4096 yerine 512x4096 olarak gncellenmiřtir.

VGG-11 ESA modeli, standart yntemle SVHN veri seti kullanılarak eęitilmiřtir. Test ve eęitim veri kmeleri iin doęruluk deęerlerinin dneme (epok) gre deęiřimi řekil 4.13.'de verilmiřtir. 20 epok boyunca eęitilen aę modeli eęitim kmesinde %98,785 ve test kmesinde %93,685 sınıflandırma doęruluęuna ulařmıřtır. Yıęın sayısı deęeri 10 olarak atanmıřtır.



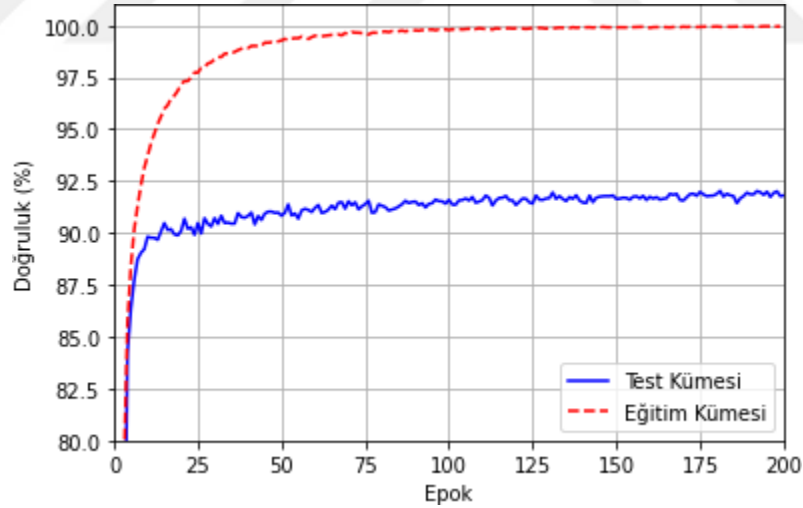
řekil 4.13. VGG-11 ESA modelinin standart yntemle SVHN veri seti ile eęitimi

VGG-11 ESA modeli, İAA yntemiyle SVHN veri seti kullanılarak eęitilmiřtir. Test ve eęitim veri kmeleri iin doęruluk deęerlerinin dneme (epok) gre deęiřimi řekil 4.14'te verilmiřtir. 50 epok boyunca eęitilen aę modeli eęitim kmesinde %99,997 ve test kmesinde %94,42 sınıflandırma doęruluęuna ulařmıřtır. Yıęın sayısı deęeri 128 olarak atanmıřtır.



Şekil 4.14. VGG-11 ESA modelinin İAA yöntemiyle SVHN veri seti kullanılarak eğitimi

VGG-11 ESA modeli, XNOR-Ağ yöntemiyle SVHN veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.15’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,962 ve test kümesinde %92,029 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

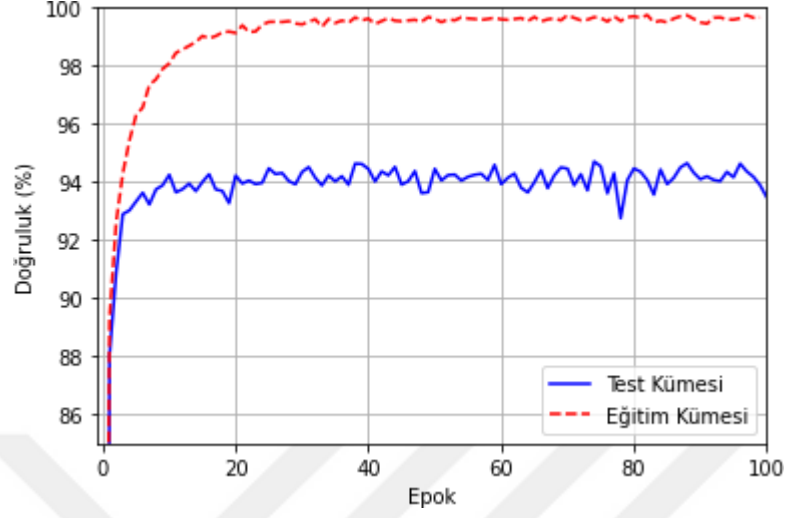


Şekil 4.15. VGG-11 ESA modelinin XNOR-Ağ yöntemiyle SVHN veri seti kullanılarak eğitimi

4.2.3. SVHN veri seti ve VGG-19-k ağ modeli ile yapılan deneyler

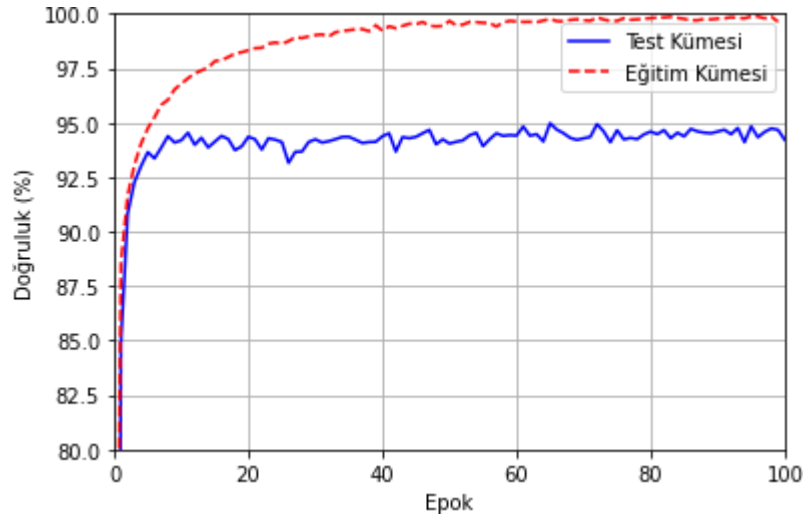
VGG-19-k ESA modeli, standart yöntemle SVHN veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.16’da verilmiştir. 100 epok boyunca eğitilen ağ modeli eğitim

kümesinde %99,749 ve test kümesinde %94,707 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



Şekil 4.16. VGG-19-k ESA modelinin standart yöntemle SVHN veri seti ile eğitimi

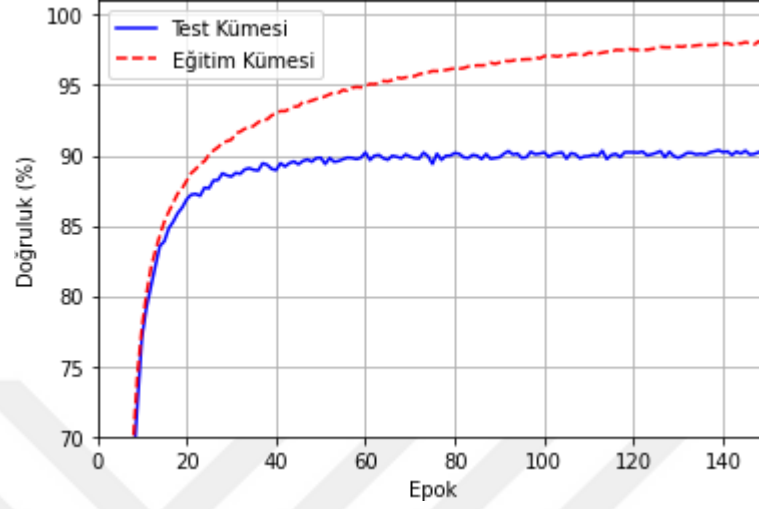
VGG-19-k ESA modeli, İAA yöntemiyle SVHN veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.17’de verilmiştir. 100 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,881 ve test kümesinde %94,975 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.17. VGG-19-k ESA modelinin İAA yöntemiyle SVHN veri seti kullanılarak eğitimi

VGG-19-k ESA modeli, XNOR-Ağ yöntemiyle SVHN veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre

değişimi Şekil 4.18’de verilmiştir. 150 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,110 ve test kümesinde %90,362 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.18. VGG-19_ESA modelinin XNOR-Ağ yöntemiyle SVHN veri seti kullanılarak eğitimi

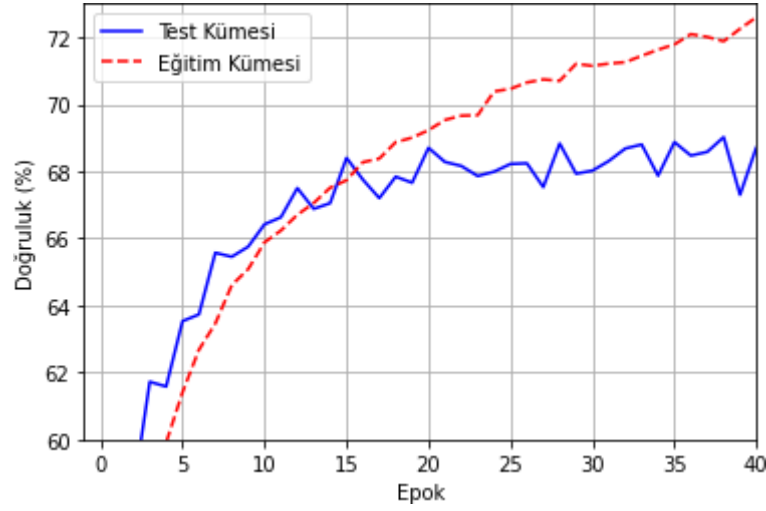
4.3. CIFAR-10 Veri Seti Üzerinde Yapılan Deneyler

CIFAR-10 veri seti hakkında bilgiler Bölüm 3.4.3’te detaylı olarak verilmiştir. Bu veri seti üzerinde LeNet-5, AlexNet, VGG-11, VGG-13, VGG-16, VGG-19, AlexNet-k, VGG-11-k, VGG-13-k, VGG-16-k ve VGG-19-k ESA modelleri denenmiştir.

4.3.1. CIFAR-10 veri seti ve LeNet-5 ağ modeli ile yapılan deneyler

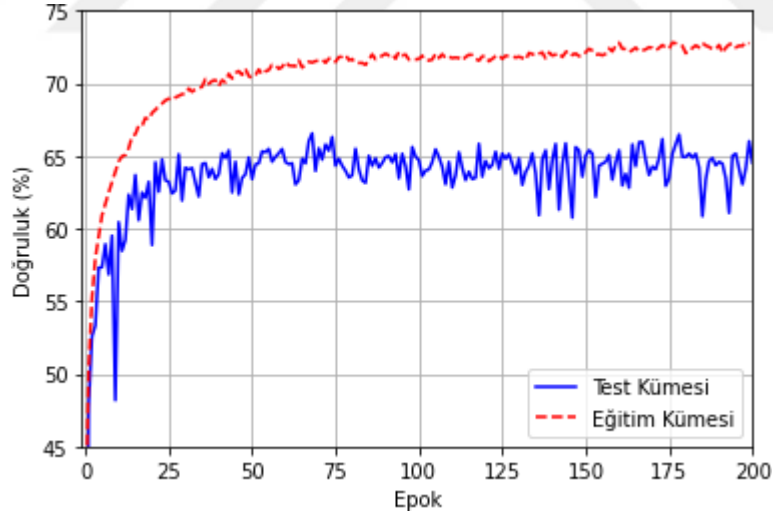
Bu başlık altında yapılan çalışmalarda, veri setinde herhangi bir ön işleme ya da veri çoğaltma yapılmamıştır. Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 1x32x32 boyutlu olduğu için olduğu için ilk evrişim katmanında filtrelerin derinliği 1’den 3’e çıkarılmıştır. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

LeNet-5 ESA modeli, standart yöntemle CIFAR-10 veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.19’da verilmiştir. 40 epok boyunca eğitilen ağ modeli eğitim kümesinde %72,252 ve test kümesinde %69,02 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



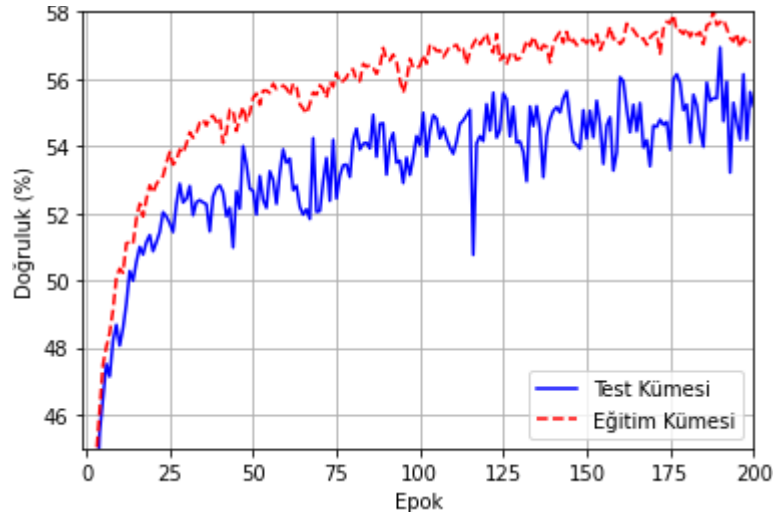
Şekil 4.19. LeNet-5 ESA modelinin standart yöntemle CIFAR-10 veri seti ile eğitimi

LeNet-5 ESA modeli, İAA yöntemiyle CIFAR-10 veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.20’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %72,85 ve test kümesinde %66,56 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.20. LeNet-5 ESA modelinin İAA yöntemiyle CIFAR-10 veri seti kullanılarak eğitimi

LeNet-5 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.21’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %58,018 ve test kümesinde %56,94 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

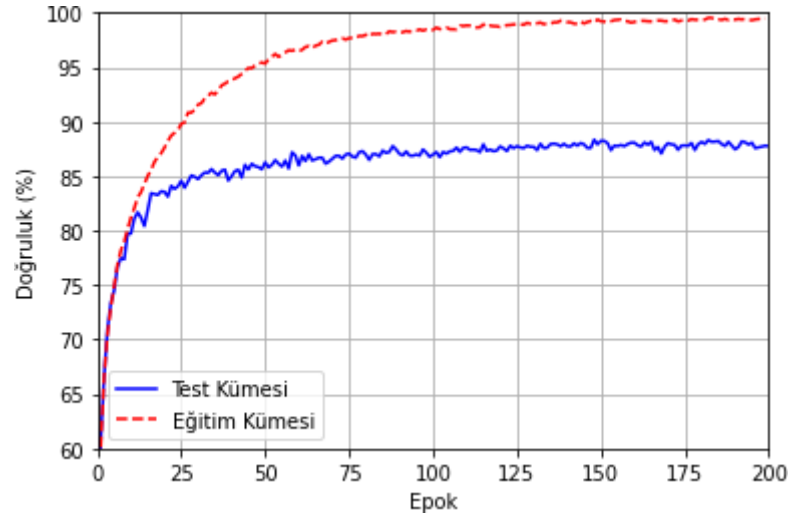


Şekil 4.21. LeNet-5 ESA modelinin XNOR-Ağ yöntemiyle CIFAR-10 veri seti kullanılarak eğitimi

4.3.2. CIFAR-10 veri seti ve AlexNet ağ modeli ile yapılan deneyler

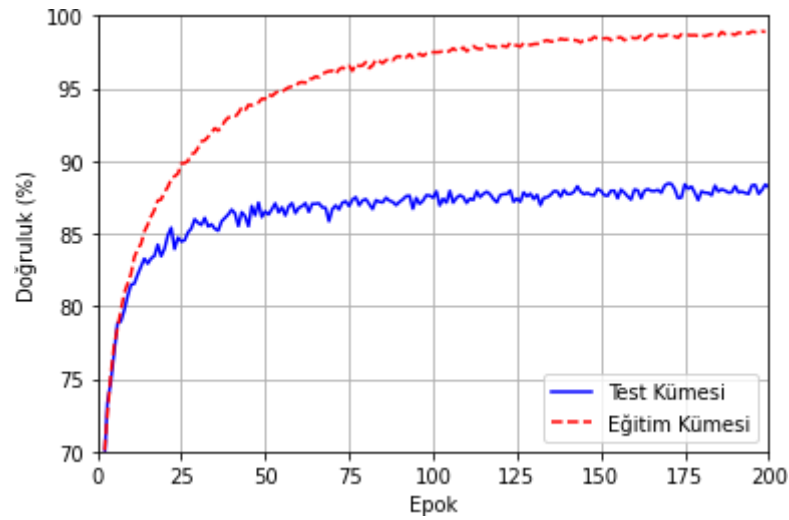
Bu başlık altında yapılan çalışmalarda, veri çoğaltma işlemi yapılarak sonuçlar elde edilmiştir. Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için ilk evrişim katmanında 11x11 filtreler ve 4 adım evrişim işlemi yerine 3x3 filtreler 1 adım evrişim işlemi yapılmıştır. Bir sonraki evrişim katmanında da 5x5 filtreler yerine 3x3 filtreler kullanılmıştır. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

AlexNet ESA modeli, standart yöntemle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.22’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,53 ve test kümesinde %88,35 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.22. AlexNet ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yöntemi ile eğitimi

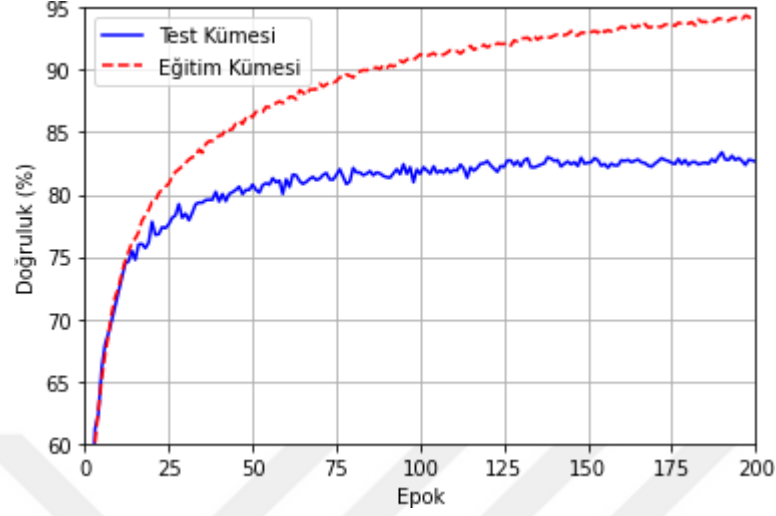
AlexNet ESA modeli, İAA yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.23’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,92 ve test kümesinde %88,45 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.23. AlexNet ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

AlexNet ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.24’te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %94,364 ve

test kümesinde %83,39 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

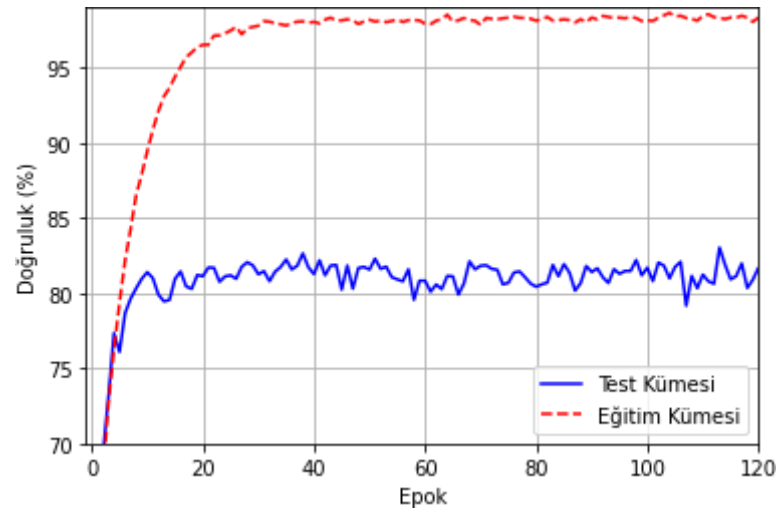


Şekil 4.24. AlexNet ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.3.3. CIFAR-10 veri seti ve VGG-11 ağ modeli ile yapılan deneyler

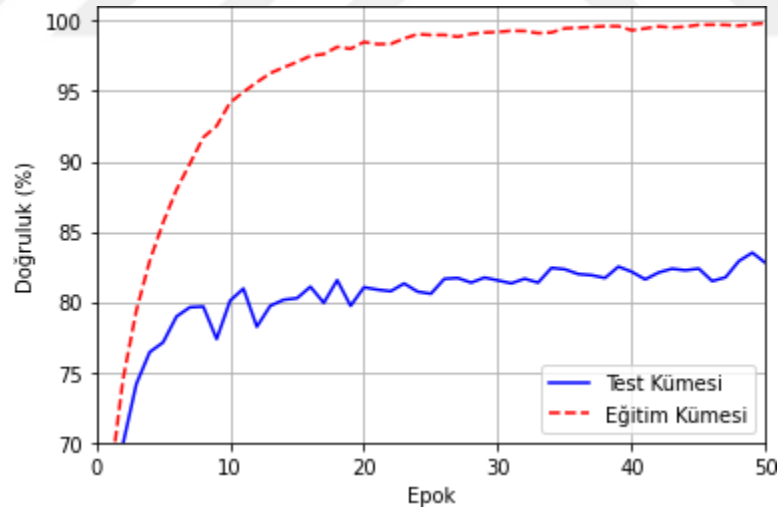
Bu başlık altında yapılan çalışmalarda, veriler hem ön işlemez olarak hem de veri çoğaltma işlemi ile çeşitlendirilerek kullanılmıştır. Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için tam bağlantı katmanına ulaşan öznitelik haritaları 7x7 olması gerekirken 1x1 olmaktadır. Bu yüzden son evrişim katmanı ile ilk tam bağlantı katmanı arasındaki bağlantı 25.088x4096 yerine 512x4096 olarak güncellenmiştir. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

VGG-11 ESA modeli, standart yöntemle CIFAR-10 veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.25'te verilmiştir. 120 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,658 ve test kümesinde %83,04 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



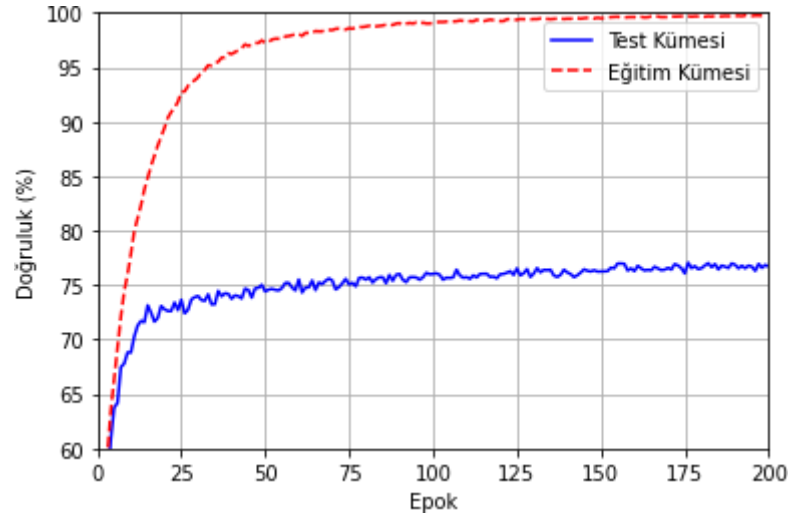
Şekil 4.25. VGG-11 ESA modelinin standart yöntemle CIFAR-10 veri seti ile eğitimi

VGG-11 ESA modeli, İAA yöntemiyle CIFAR-10 veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.26’da verilmiştir. 130 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,756 ve test kümesinde %83,5 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



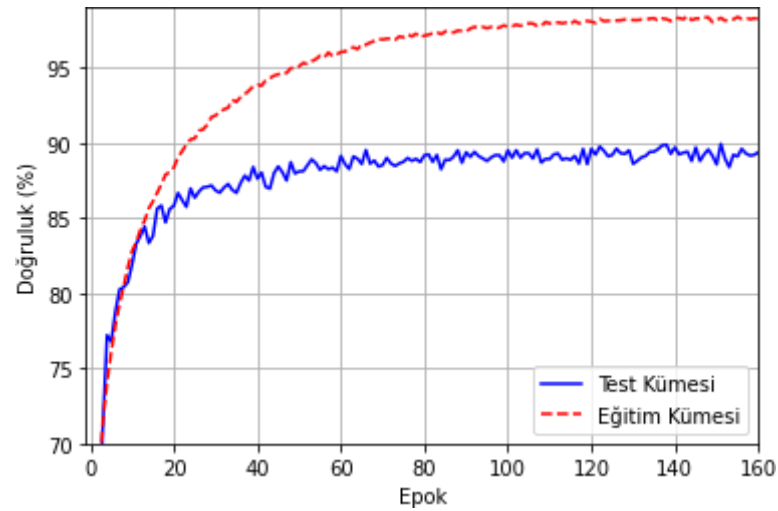
Şekil 4.26. VGG-11 ESA modelinin İAA yöntemiyle CIFAR-10 veri seti kullanılarak eğitimi

VGG-11 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.27’de verilmiştir. 100 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,758 ve test kümesinde %77,06 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



Şekil 4.27. VGG-11 ESA modelinin XNOR-Ağ yöntemiyle CIFAR-10 veri seti kullanılarak eğitimi

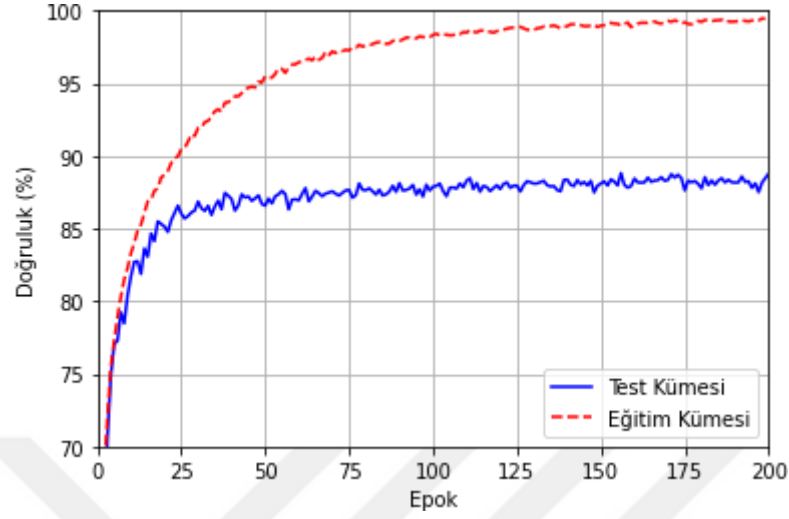
VGG-11 ESA modeli, standart yöntemle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.28’de verilmiştir. 160 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,458 ve test kümesinde %89,94 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



Şekil 4.28. VGG-11 ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yöntemi ile eğitimi

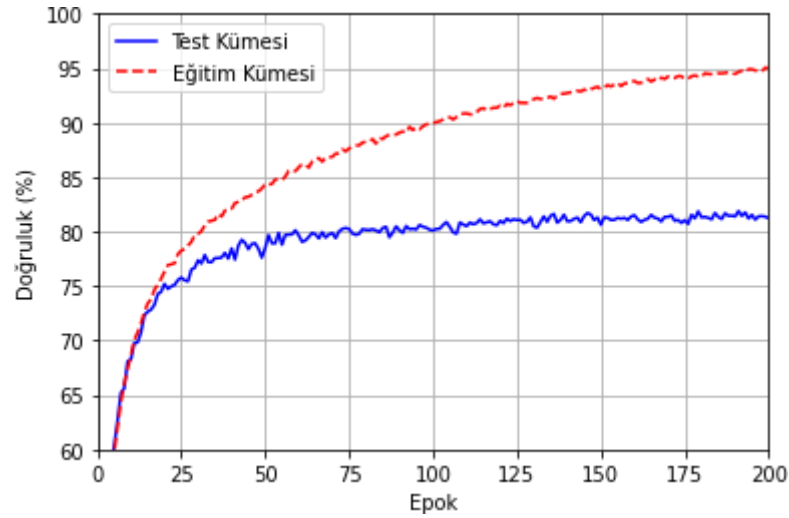
VGG-11 ESA modeli, İAA yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.29’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,468 ve

test kümesinde %88,82 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.29. VGG-11 ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-11 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.30’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %95,312 ve test kümesinde %82,1 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

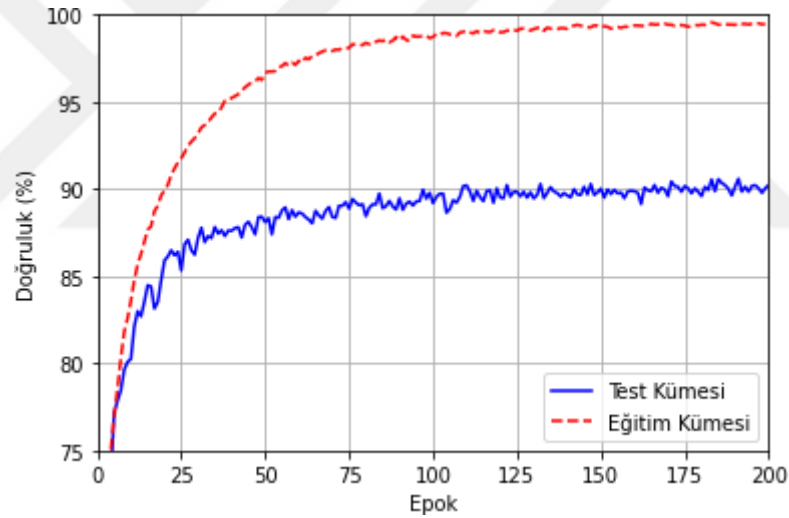


Şekil 4.30. VGG-11 ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.3.4. CIFAR-10 veri seti ve VGG-13 ağ modeli ile yapılan deneyler

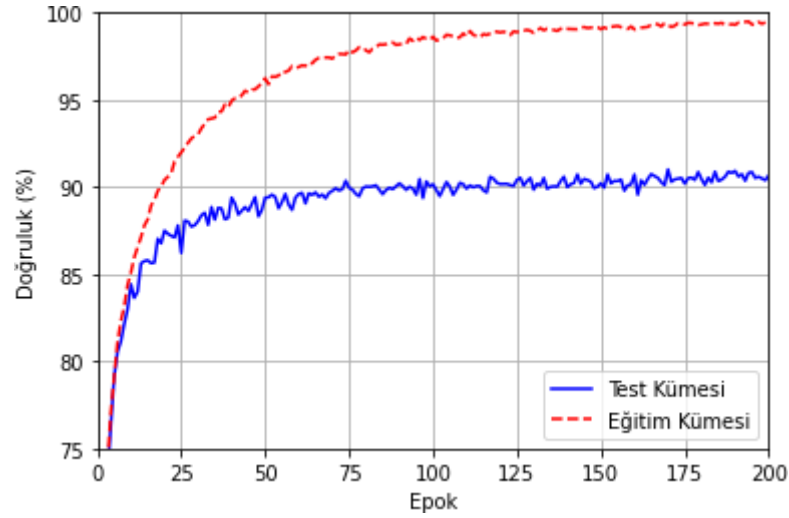
Bu başlık altında yapılan çalışmalarda, veri çoğaltma işlemi yapılarak sonuçlar elde edilmiştir. Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için tam bağlantı katmanına ulaşan öznetelik haritaları 7x7 olması gerekirken 1x1 olmaktadır. Bu yüzden son evrişim katmanı ile ilk tam bağlantı katmanı arasındaki bağlantı 25.088x4096 yerine 512x4096 olarak güncellenmiştir. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

VGG-13 ESA modeli, standart yöntemle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.31’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,592 ve test kümesinde %90,59 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



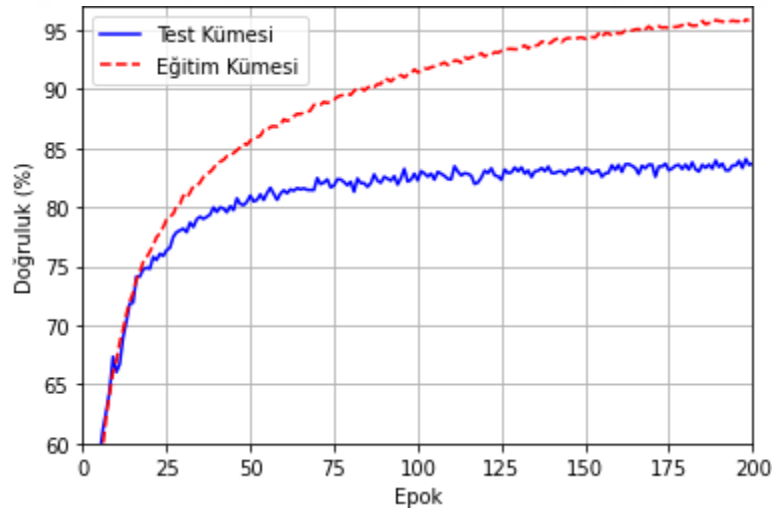
Şekil 4.31. VGG-13 ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yöntemi ile eğitimi

VGG-13 ESA modeli, İAA yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.32’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %91,042 ve test kümesinde %91,01 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



Şekil 4.32. VGG-13 ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-13 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.33'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %95, 908 ve test kümesinde %84,1 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



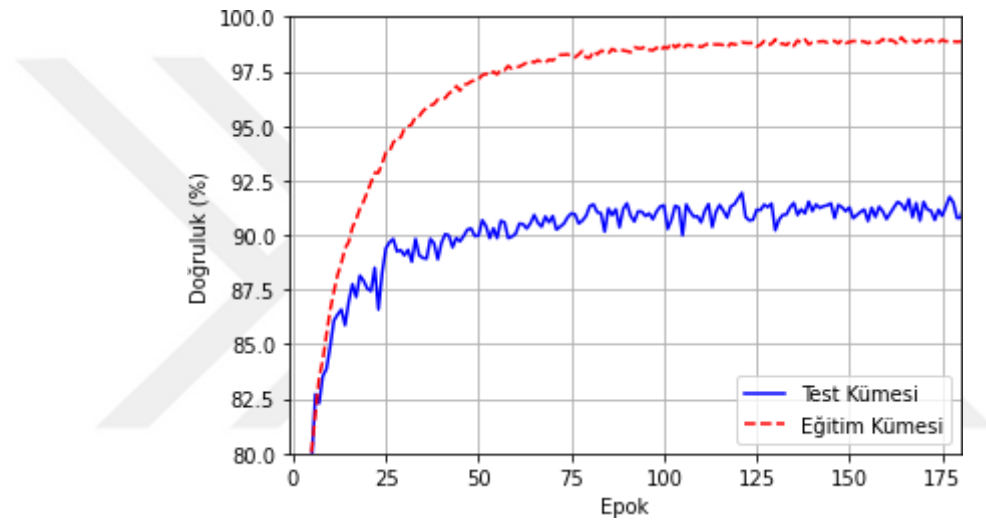
Şekil 4.33. VGG-13 ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.3.5. CIFAR-10 veri seti ve VGG-16 ağ modeli ile yapılan deneyler

Bu başlık altında yapılan çalışmalarda, veri çoğaltma işlemi yapılarak sonuçlar elde edilmiştir. Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için tam bağlantı katmanına ulaşan öznetelik haritaları 7x7 olması gerekirken 1x1

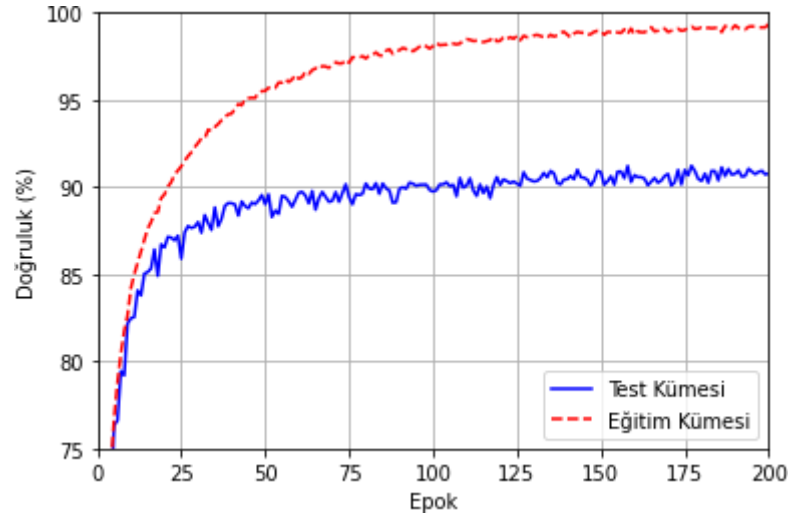
olmaktadır. Bu yüzden son evriřim katmanı ile ilk tam baęlantı katmanı arasındaki baęlantı 25.088x4096 yerine 512x4096 olarak gncellenmiřtir. Aktivasyon fonksiyonu olarak ReLU kullanılmıřtır.

VGG-16 ESA modeli, standart yntemle CIFAR-10 veri seti ve veri artırma yntemlerinden evirme ve rastgele kırpma yntemleri kullanılarak eęitilmiřtir. Test ve eęitim veri kmeleri iin doęruluk deęerlerinin dneme (epok) gre deęiřimi řekil 4.34'te verilmiřtir. 180 epok boyunca eęitilen aę modeli eęitim kmesinde %99,056 ve test kmesinde %91,93 sınıflandırma doęruluęuna ulařmıřtır. Yıęın sayısı deęeri 32 olarak atanmıřtır.



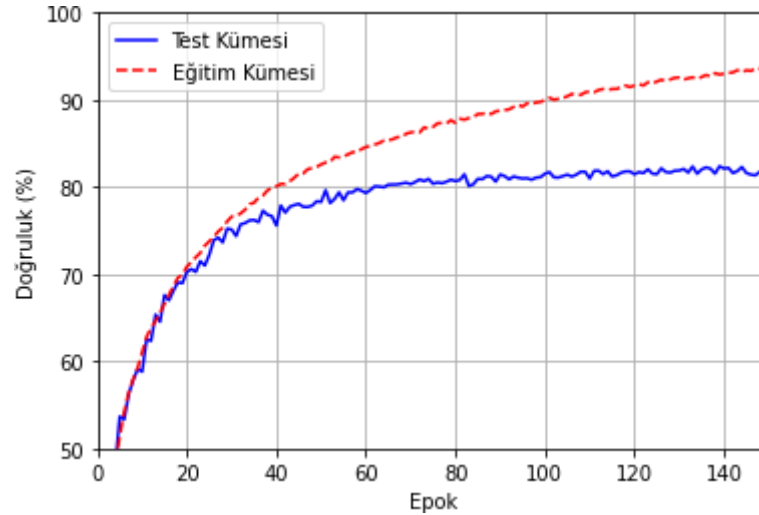
řekil 4.34. VGG-16 ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yntemi ile eęitimi

VGG-16 ESA modeli, İAA yntemiyle CIFAR-10 veri seti ve veri artırma yntemlerinden evirme ve rastgele kırpma yntemleri kullanılarak eęitilmiřtir. Test ve eęitim veri kmeleri iin doęruluk deęerlerinin dneme (epok) gre deęiřimi řekil 4.35'te verilmiřtir. 200 epok boyunca eęitilen aę modeli eęitim kmesinde %90,354 ve test kmesinde %91,22 sınıflandırma doęruluęuna ulařmıřtır. Yıęın sayısı deęeri 128 olarak atanmıřtır.



Şekil 4.35. VGG-16 ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-16 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.36’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %95,548 ve test kümesinde %82,82 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



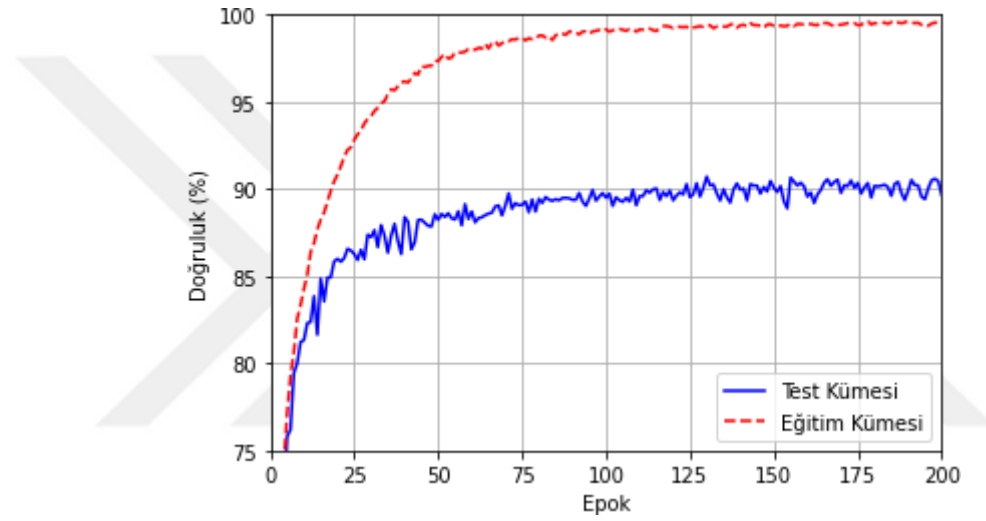
Şekil 4.36. VGG-16 ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.3.6. CIFAR-10 veri seti ve VGG-19 ağ modeli ile yapılan deneyler

Bu başlık altında yapılan çalışmalarda, veri çoğaltma işlemi yapılarak sonuçlar elde edilmiştir. Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu

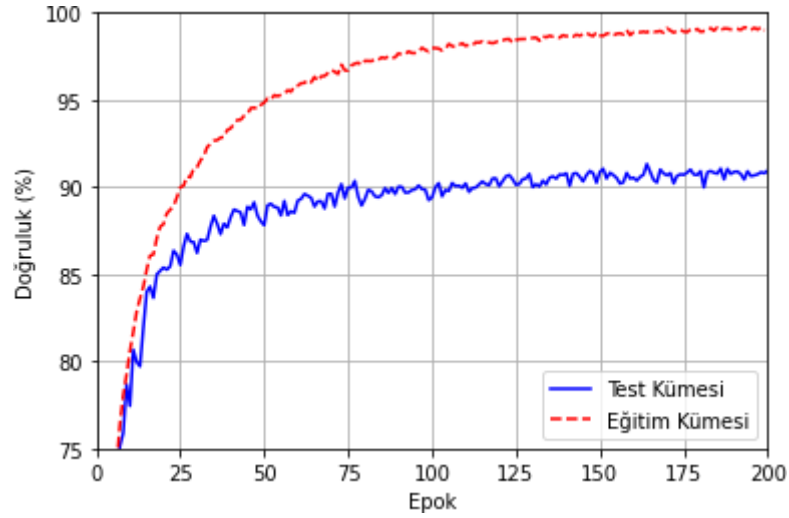
olduğu için tam bağlantı katmanına ulaşan öznetelik haritaları 7x7 olması gerekirken 1x1 olmaktadır. Bu yüzden son evrişim katmanı ile ilk tam bağlantı katmanı arasındaki bağlantı 25.088x4096 yerine 512x4096 olarak güncellenmiştir. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

VGG-19 ESA modeli, standart yöntemle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.37’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,592 ve test kümesinde %90,71 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



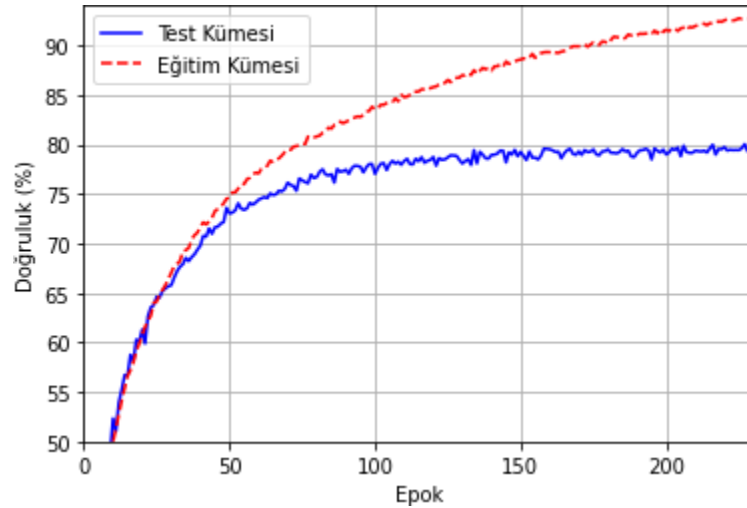
Şekil 4.37. VGG-19 ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yöntemi ile eğitimi

VGG-19 ESA modeli, İAA yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.38’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,182 ve test kümesinde %91,32 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.38. VGG-19 ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-19 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.39’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %92,732 ve test kümesinde %80,03 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

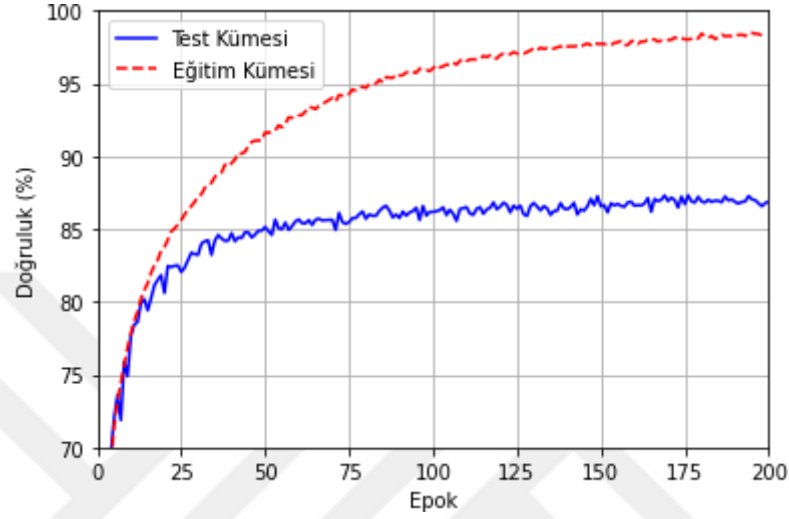


Şekil 4.39. VGG-19 ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.3.7. CIFAR-10 veri seti ve AlexNet-k ağ modeli ile yapılan deneyler

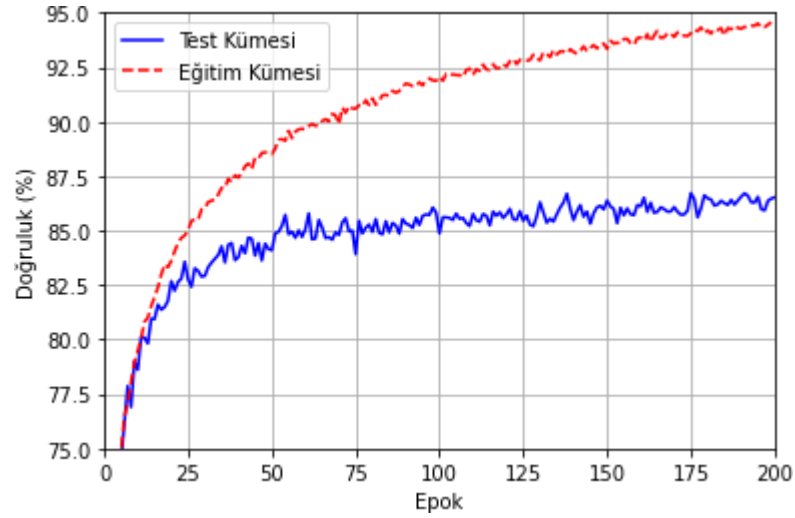
Bu başlık altında yapılan çalışmalarda, veri çoğaltma işlemi yapılarak sonuçlar elde edilmiştir. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

AlexNet-k ESA modeli, standart yöntemle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.40’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,482 ve test kümesinde %87,32 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



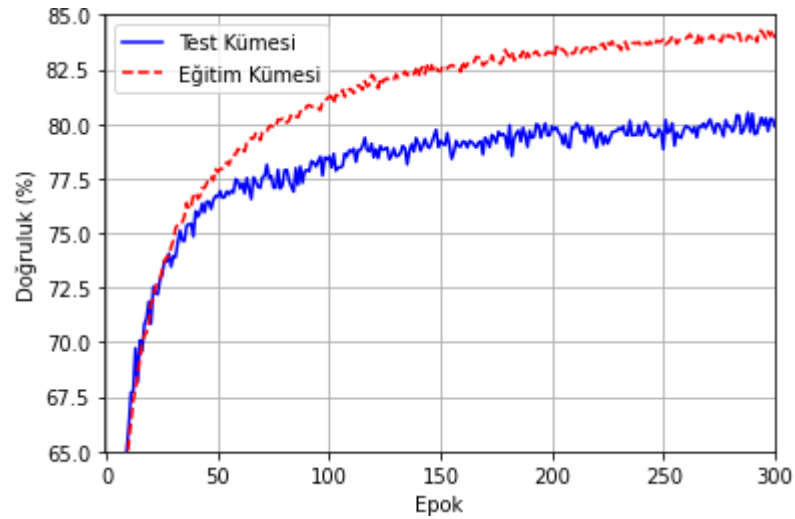
Şekil 4.40. AlexNet-k ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yöntemi ile eğitimi

AlexNet-k ESA modeli, İAA yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.41’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %94,558 ve test kümesinde %86,72 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.41. AlexNet-k ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

AlexNet-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.42’de verilmiştir. 300 epok boyunca eğitilen ağ modeli eğitim kümesinde %84,28 ve test kümesinde %80,21 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

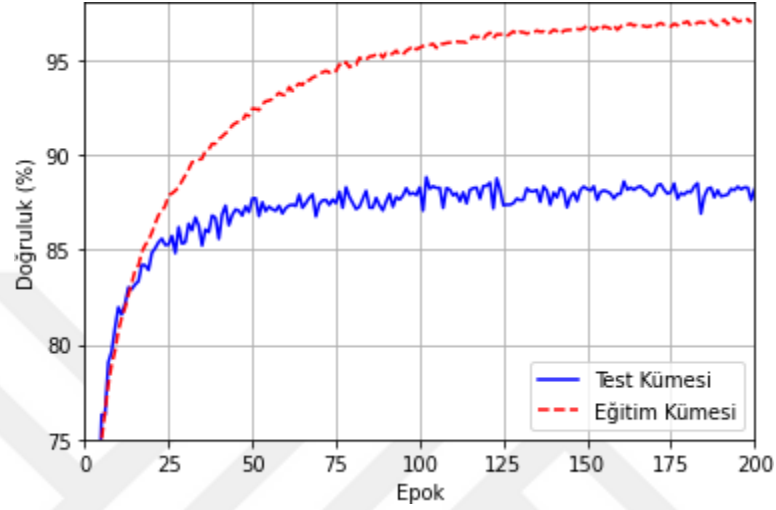


Şekil 4.42. AlexNet-k ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.3.8. CIFAR-10 veri seti ve VGG-11-k ağ modeli ile yapılan deneyler

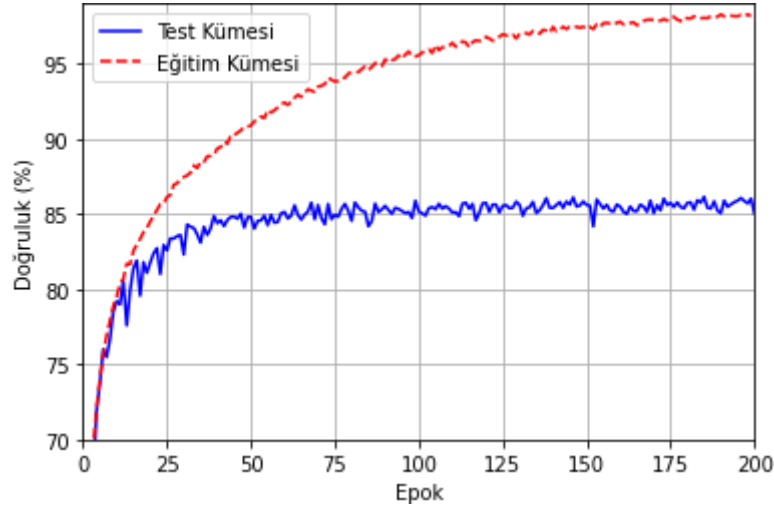
Bu başlık altında yapılan çalışmalarda, veri çoğaltma işlemi yapılarak sonuçlar elde edilmiştir. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

VGG-11-k ESA modeli, standart yöntemle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.43'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,458 ve test kümesinde %88,81 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



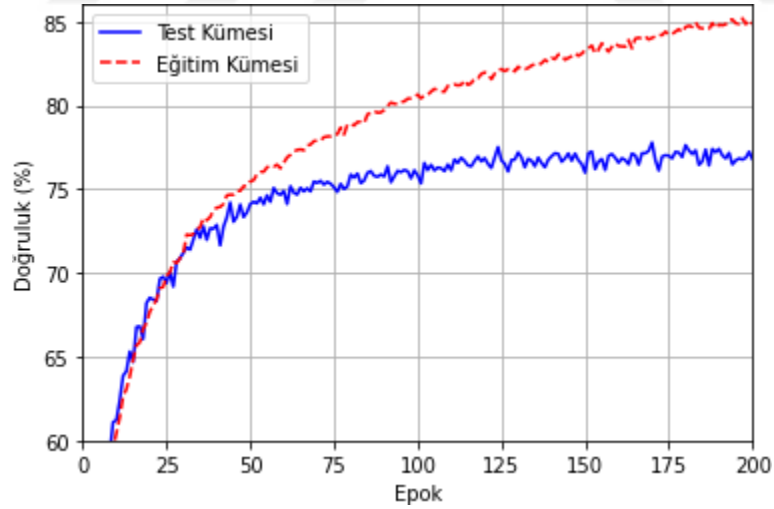
Şekil 4.43. VGG-11-k ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yöntemi ile eğitimi

VGG-11-k ESA modeli, İAA yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.44'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,294 ve test kümesinde %86,15 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.44. VGG-11-k ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-11-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.45'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %85,188 ve test kümesinde %77,78 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

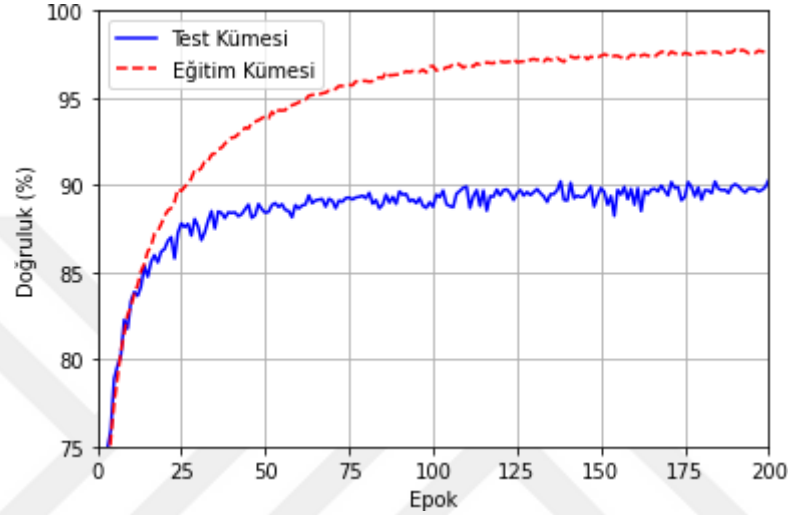


Şekil 4.45. VGG-11-k ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.3.9. CIFAR-10 veri seti ve VGG-13-k ağ modeli ile yapılan deneyler

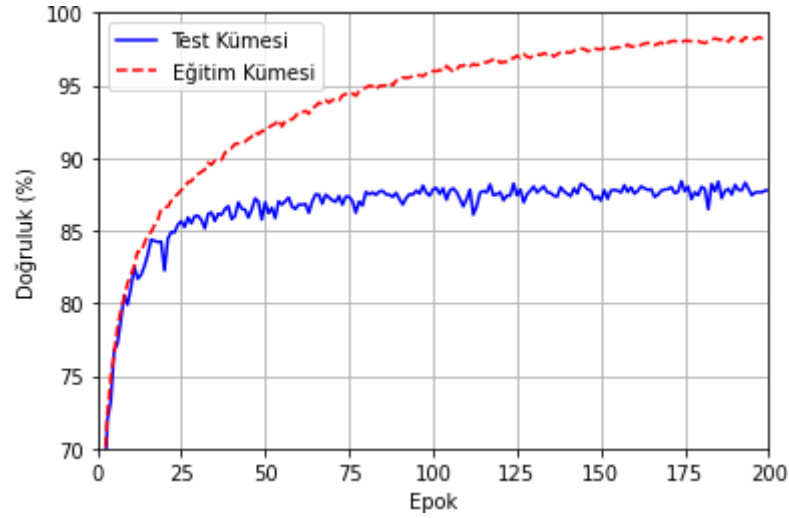
Bu başlık altında yapılan çalışmalarda, veri çoğaltma işlemi yapılarak sonuçlar elde edilmiştir. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

VGG-13-k ESA modeli, standart yöntemle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.46’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,944 ve test kümesinde %90,35 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



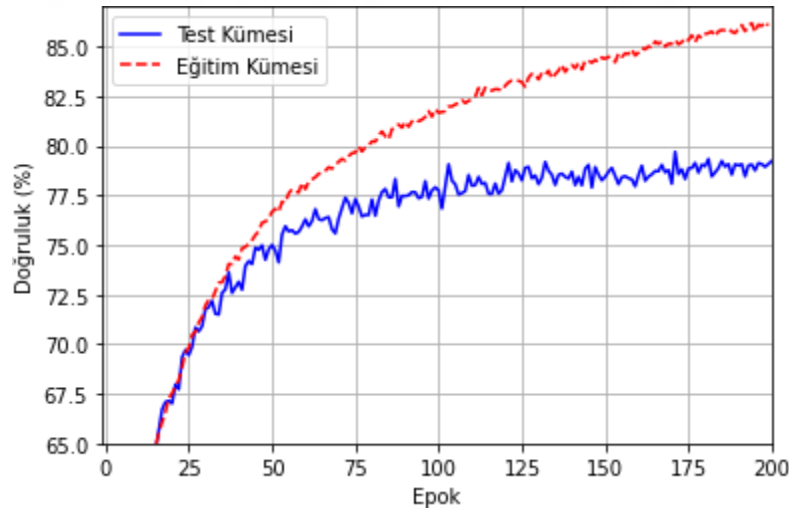
Şekil 4.46. VGG-13-k ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yöntemi ile eğitimi

VGG-13-k ESA modeli, İAA yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.47’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,354 ve test kümesinde %88,4 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.47. VGG-13-k ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-13-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpmaya yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.48’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %86,236 ve test kümesinde %79,69 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

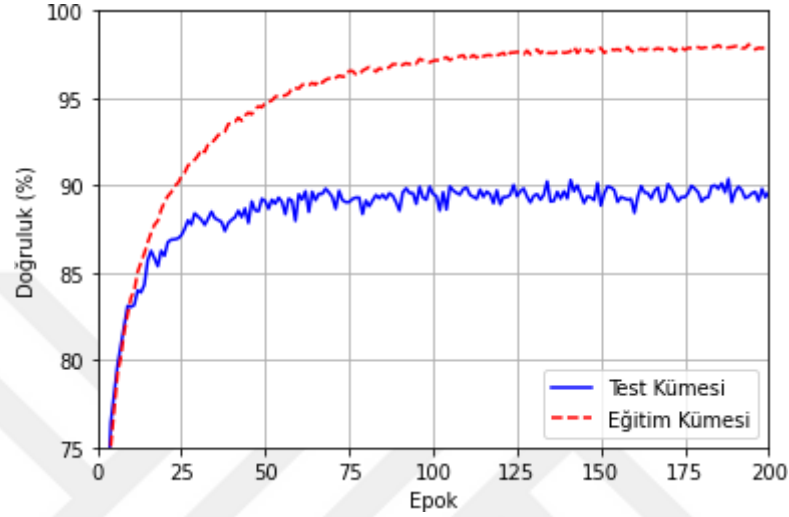


Şekil 4.48. VGG-13-k ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.3.10. CIFAR-10 veri seti ve VGG-16-k ağ modeli ile yapılan deneyler

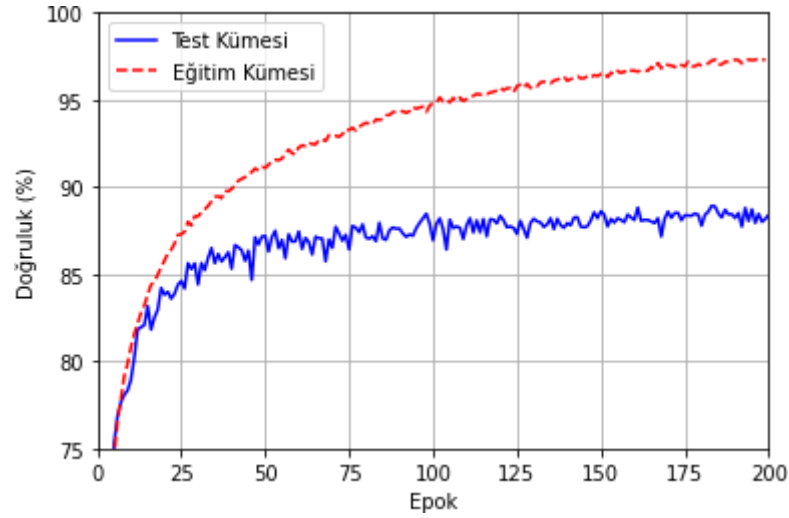
Bu başlık altında yapılan çalışmalarda, veri çoğaltma işlemi yapılarak sonuçlar elde edilmiştir. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

VGG-16-k ESA modeli, standart yöntemle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.49’da verilmiştir. 180 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,072 ve test kümesinde %90,37 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



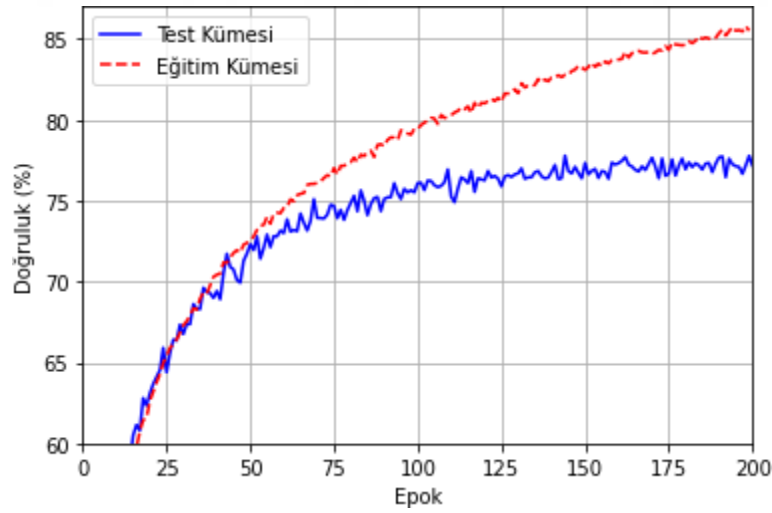
Şekil 4.49. VGG-16-k ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yöntemi ile eğitimi

VGG-16-k ESA modeli, İAA yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.50’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %97,368 ve test kümesinde %88,92 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.50. VGG-16-k ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-16-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.51’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %85,796 ve test kümesinde %77,79 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

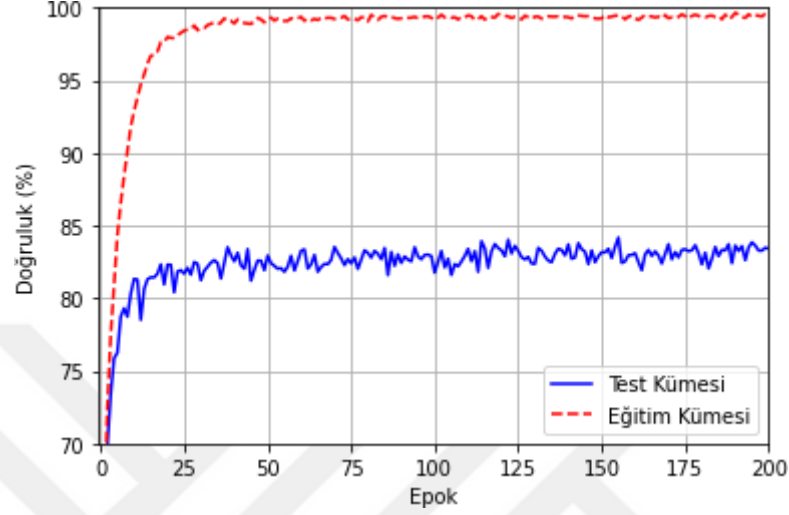


Şekil 4.51. VGG-16-k ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.3.11. CIFAR-10 veri seti ve VGG-19-k ağ modeli ile yapılan deneyler

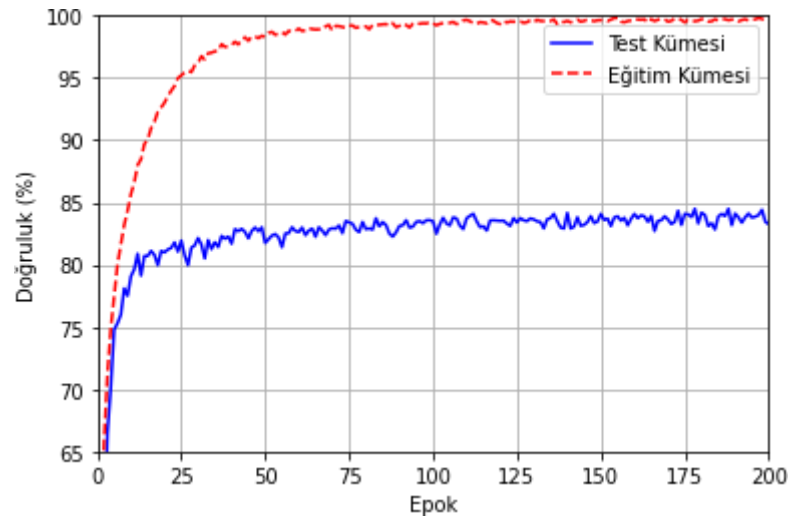
Bu başlık altında yapılan çalışmalarda, veriler hem ön işlemez olarak hem de veri çoğaltma işlemi ile çeşitlendirilerek kullanılmıştır. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

VGG-19-k ESA modeli, standart yöntemle CIFAR-10 veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.52’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,65 ve test kümesinde %84,18 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



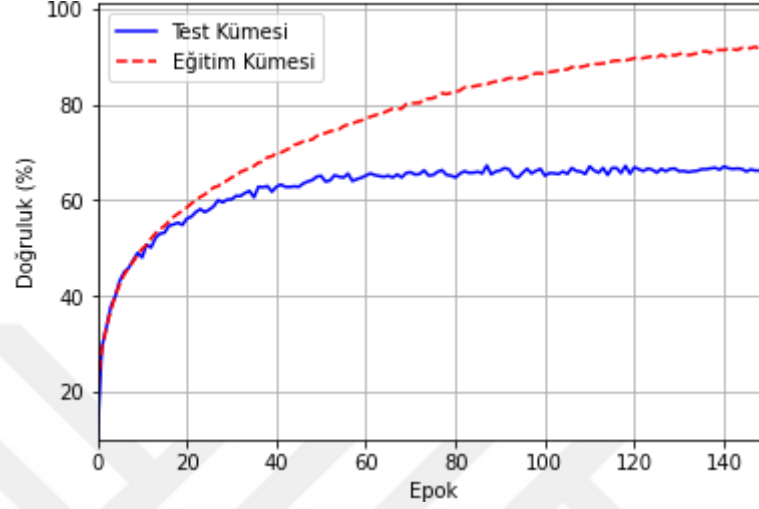
Şekil 4.52. VGG-19-k ESA modelinin standart yöntemle CIFAR-10 veri seti ile eğitimi

VGG-19-k ESA modeli, İAA yöntemiyle CIFAR-10 veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.53’te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,832 ve test kümesinde %84,49 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



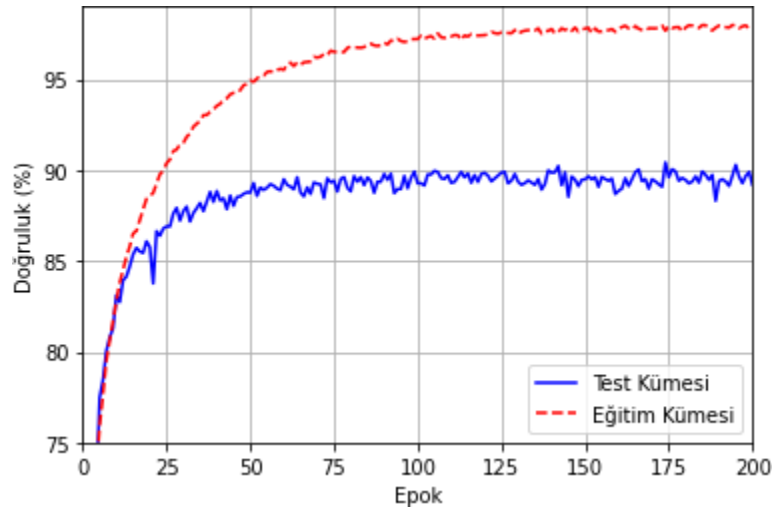
Şekil 4.53. VGG-19-k ESA modelinin İAA yöntemiyle CIFAR-10 veri seti kullanılarak eğitimi

VGG-19-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.54'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %93,302 ve test kümesinde %67,23 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



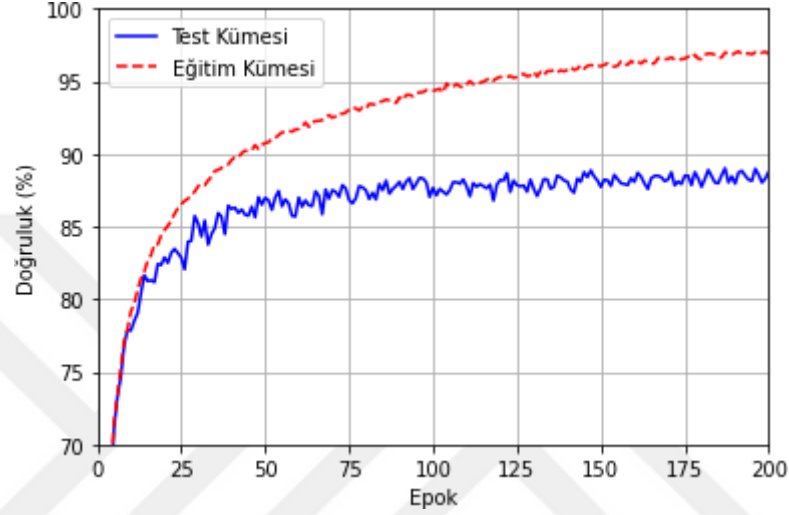
Şekil 4.54. VGG-19-k ESA modelinin XNOR-Ağ yöntemiyle CIFAR-10 veri seti kullanılarak eğitimi

VGG-19-k ESA modeli, standart yöntemle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.55'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %97,994 ve test kümesinde %90,43 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



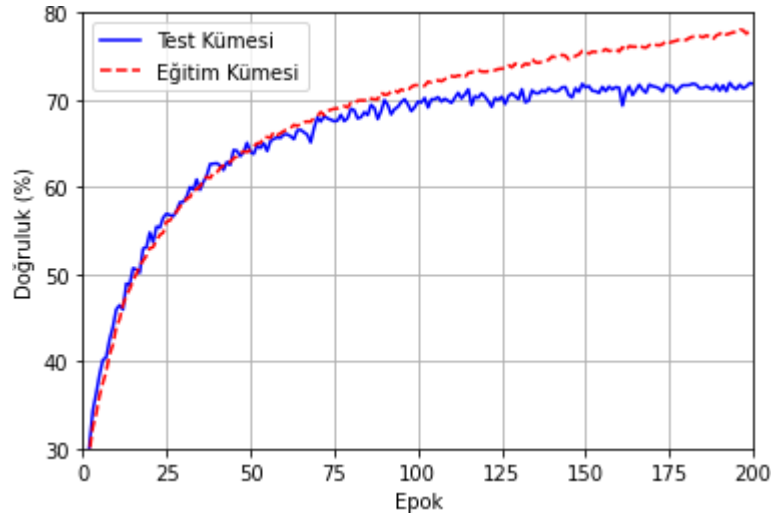
Şekil 4.55. VGG-19-k ESA modelinin CIFAR-10 veri seti kullanılarak veri artırma yöntemi ile eğitimi

VGG-19-k ESA modeli, İAA yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.56’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %97,068 ve test kümesinde %89,02 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.56. VGG-19-k ESA modelinin CIFAR-10 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-19-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-10 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.57’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %78,066 ve test kümesinde %71,96 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.57. VGG-16-k ESA modelinin CIFAR-10 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.4. CIFAR-100 Veri Seti Üzerinde Yapılan Deneyler

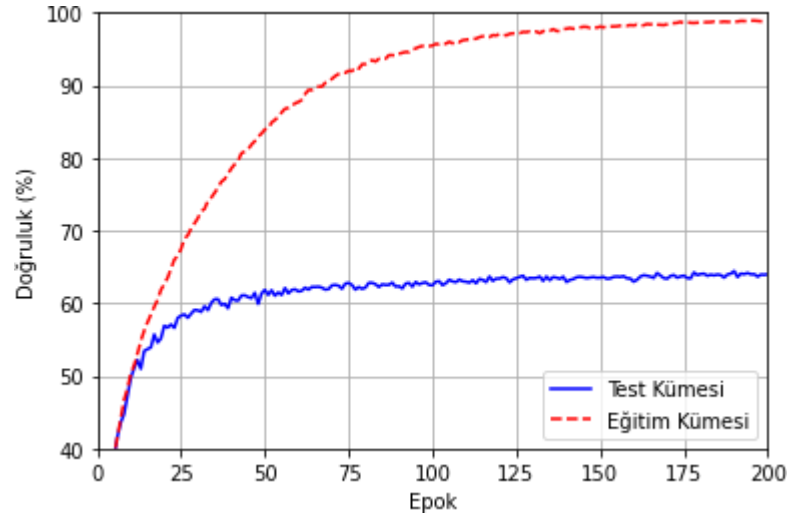
CIFAR-100 veri seti hakkında bilgiler Bölüm 3.4.4'te detaylı olarak verilmiştir. Bu veri seti üzerinde AlexNet, VGG-11, VGG-13, VGG-16, VGG-19, AlexNet-k, VGG-11-k, VGG-13-k, VGG-16-k ve VGG-19-k ESA modelleri denenmiştir.

Bu başlık altında yapılan çalışmalarda, veri çoğaltma işlemi kullanılarak sonuçlar elde edilmiştir. Aktivasyon fonksiyonu olarak ReLU kullanılmıştır.

4.4.1. CIFAR-100 veri seti ve AlexNet ağ modeli ile yapılan deneyler

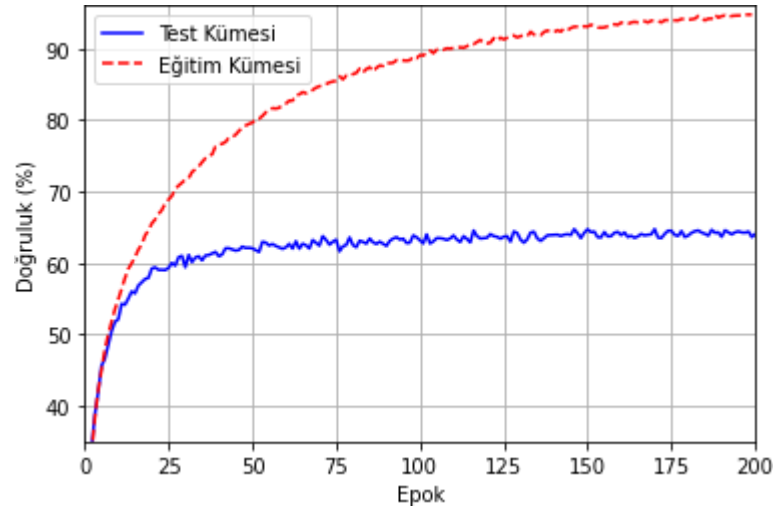
Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için ilk evrişim katmanında 11x11 filtreler ve 4 adım evrişim işlemi yerine 3x3 filtreler 1 adım evrişim işlemi yapılmıştır. Bir sonraki evrişim katmanında da 5x5 filtreler yerine 3x3 filtreler kullanılmıştır.

AlexNet ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.58'de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %98,916 ve test kümesinde %64,43 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.58. AlexNet ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

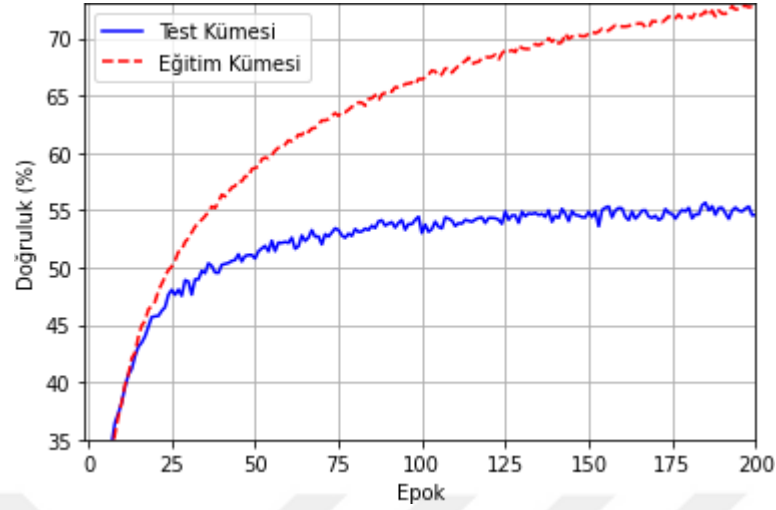
AlexNet ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.59’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %94,758 ve test kümesinde %64,73 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.59. AlexNet ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

AlexNet ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.60’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %72,812 ve

test kümesinde %55,65 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

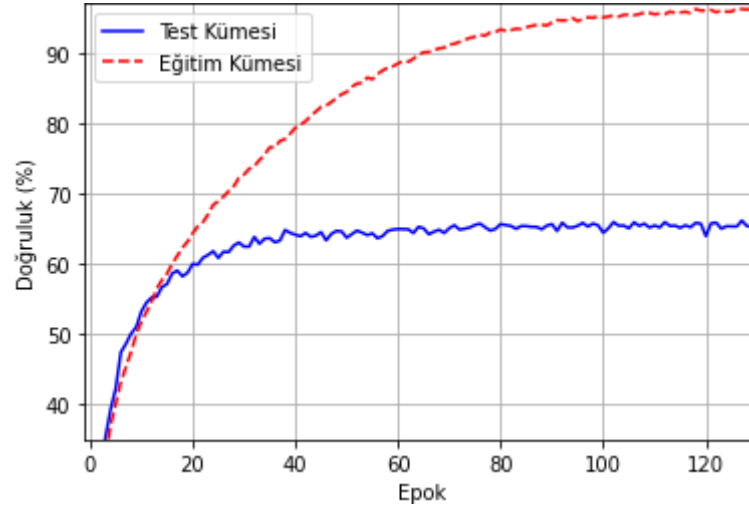


Şekil 4.60. AlexNet ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.4.2. CIFAR-100 veri seti ve VGG-11 ağ modeli ile yapılan deneyler

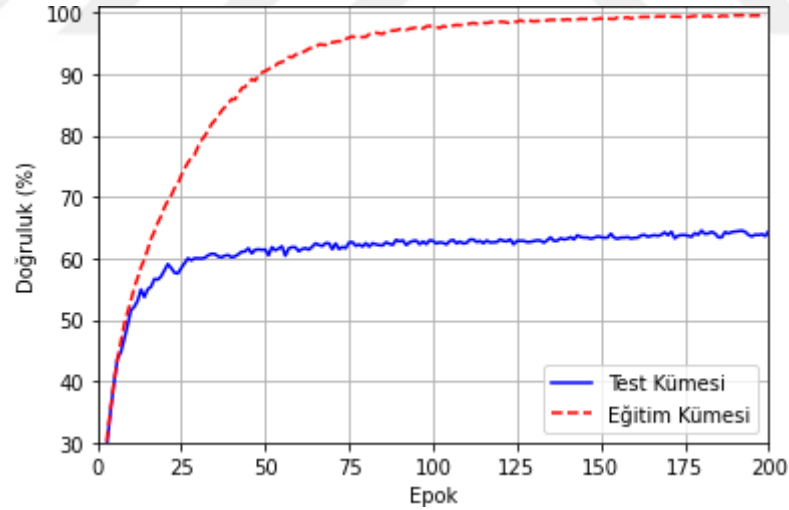
Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için tam bağlantı katmanına ulaşan öznitelik haritaları 7x7 olması gerekirken 1x1 olmaktadır. Bu yüzden son evrişim katmanı ile ilk tam bağlantı katmanı arasındaki bağlantı 25.088x4096 yerine 512x4096 olarak güncellenmiştir.

VGG-11 ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.61’de verilmiştir. 130 epok boyunca eğitilen ağ modeli eğitim kümesinde %96,338 ve test kümesinde %66,14 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 10 olarak atanmıştır.



Şekil 4.61. VGG-11 ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

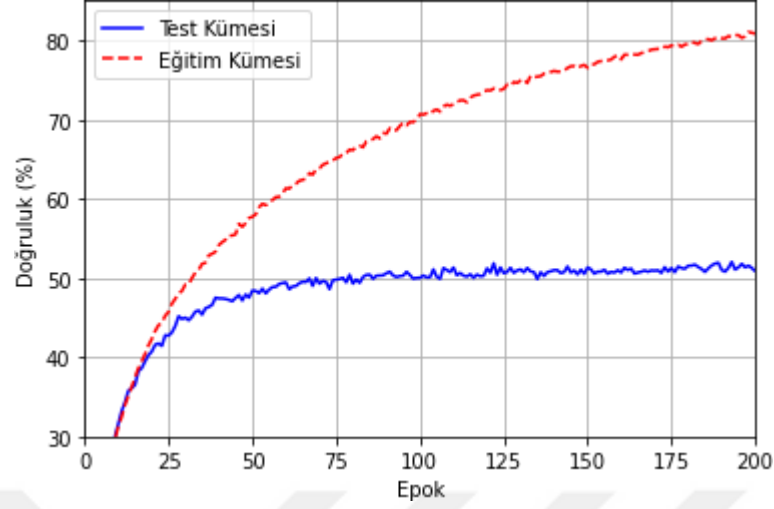
VGG-11 ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.62’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,636 ve test kümesinde %64,52 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.62. VGG-11 ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-11 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.63’te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %81,042 ve

test kümesinde %52,03 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

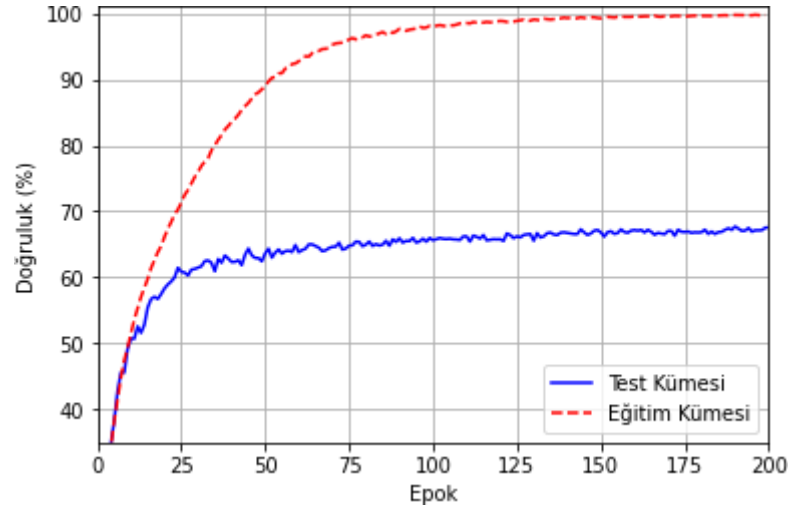


Şekil 4.63. VGG-11 ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.4.3. CIFAR-100 veri seti ve VGG-13 ağ modeli ile yapılan deneyler

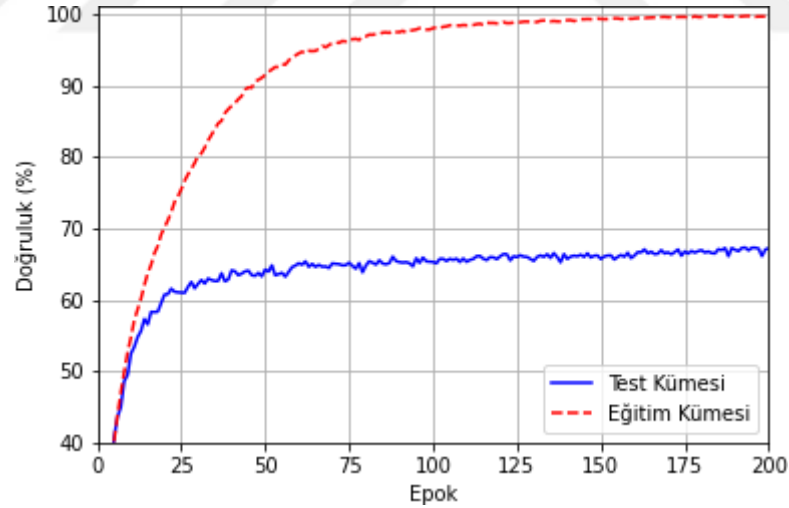
Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için tam bağlantı katmanına ulaşan öznitelik haritaları 7x7 olması gerekirken 1x1 olmaktadır. Bu yüzden son evrişim katmanı ile ilk tam bağlantı katmanı arasındaki bağlantı 25.088x4096 yerine 512x4096 olarak güncellenmiştir.

VGG-13 ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.64'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,784 ve test kümesinde %67,71 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.64. VGG-13 ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

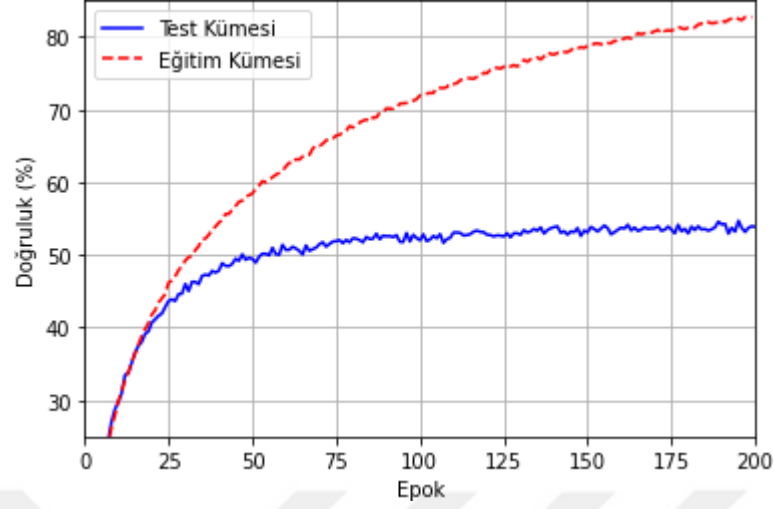
VGG-13 ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.65'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,774 ve test kümesinde %67,29 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.65. VGG-13 ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-13 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.66'da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %82,764 ve

test kümesinde %54,7 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

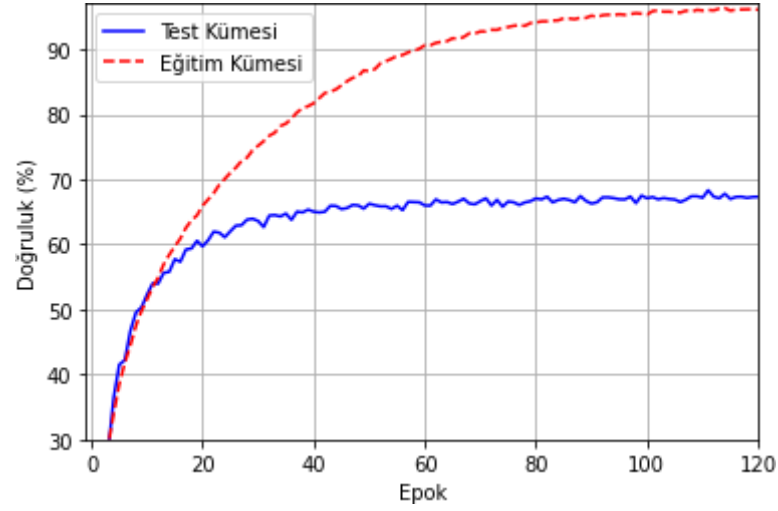


Şekil 4.66. VGG-13 ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.4.4. CIFAR-100 veri seti ve VGG-16 ağ modeli ile yapılan deneyler

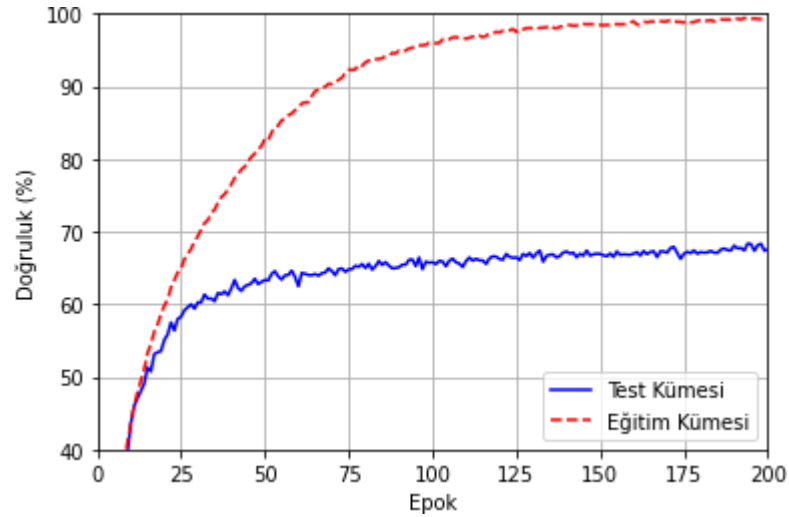
Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için tam bağlantı katmanına ulaşan öznitelik haritaları 7x7 olması gerekirken 1x1 olmaktadır. Bu yüzden son evrişim katmanı ile ilk tam bağlantı katmanı arasındaki bağlantı 25.088x4096 yerine 512x4096 olarak güncellenmiştir.

VGG-16 ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.67’de verilmiştir. 120 epok boyunca eğitilen ağ modeli eğitim kümesinde %96,318 ve test kümesinde %68,32 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



Şekil 4.67. VGG-16 ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

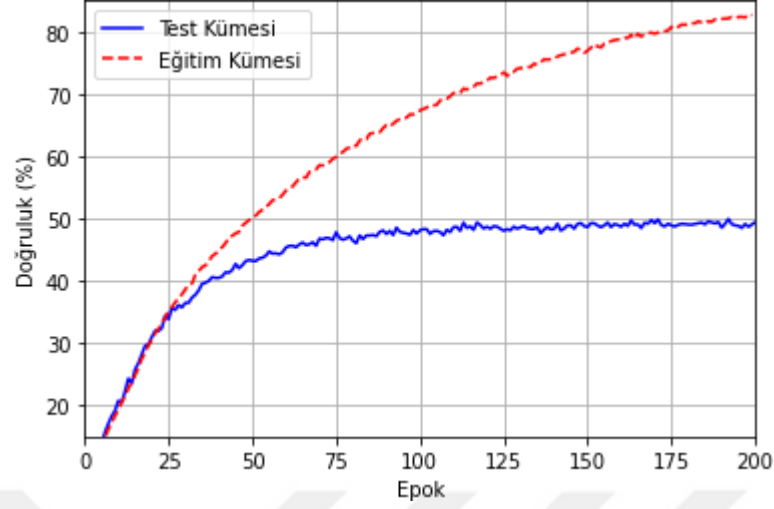
VGG-16 ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.68’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,418 ve test kümesinde %68,34 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.68. VGG-16 ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-16 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.69’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %82,834 ve

test kümesinde %49,95 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

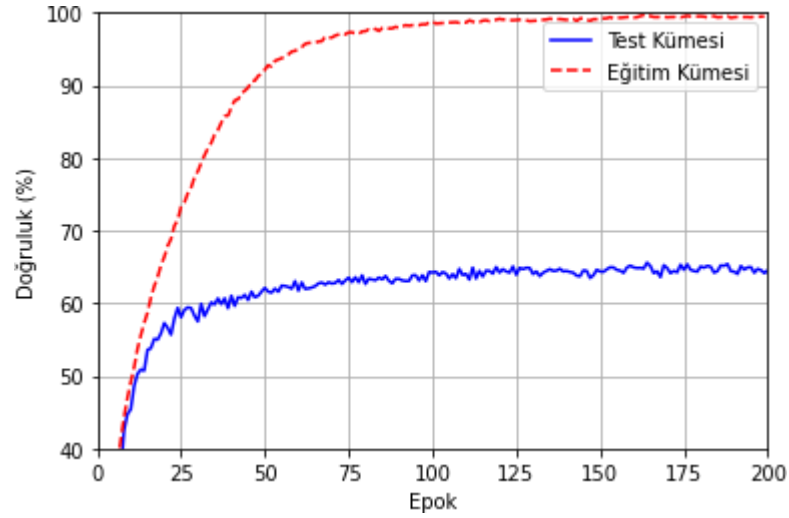


Şekil 4.69. VGG-16 ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.4.5. CIFAR-100 veri seti ve VGG-19 ağ modeli ile yapılan deneyler

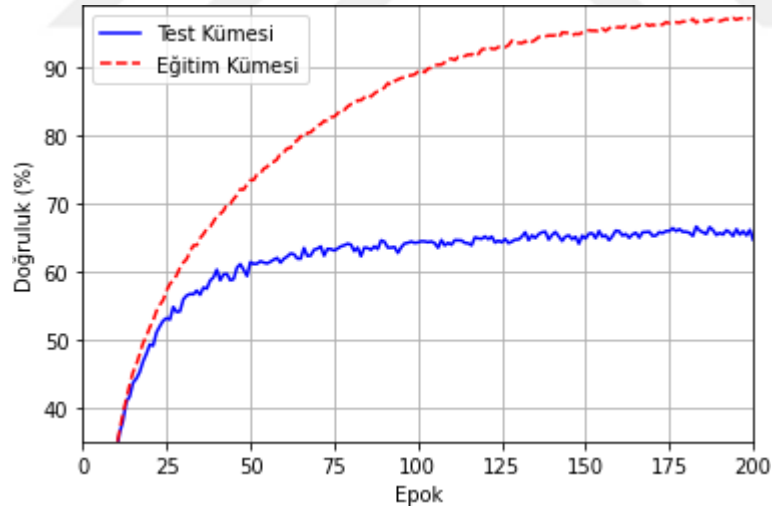
Veri seti 3x32x32 boyutlu ve ağ modelinin girişi 3x224x224 boyutlu olduğu için tam bağlantı katmanına ulaşan öznitelik haritaları 7x7 olması gerekirken 1x1 olmaktadır. Bu yüzden son evrişim katmanı ile ilk tam bağlantı katmanı arasındaki bağlantı 25.088x4096 yerine 512x4096 olarak güncellenmiştir.

VGG-19 ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.70’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %99,634 ve test kümesinde %65,59 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.70. VGG-19 ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

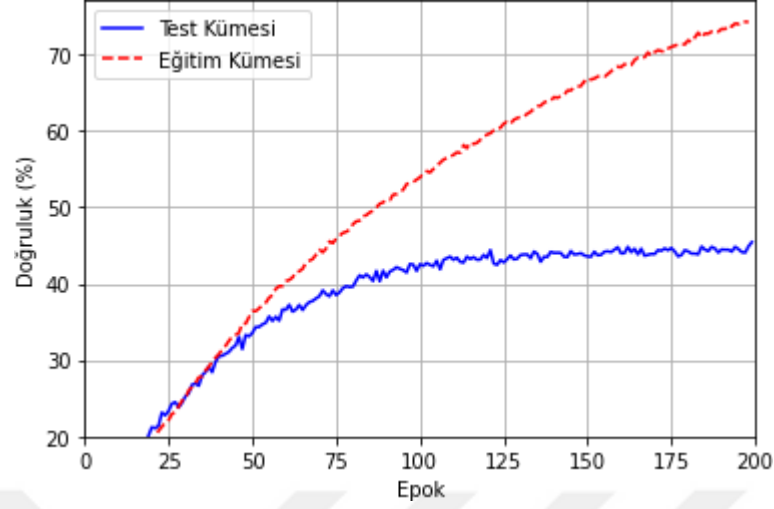
VGG-19 ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpmaya yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.71’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %97,294 ve test kümesinde %66,57 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.71. VGG-19 ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-19 ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpmaya yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.72’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %74,244 ve

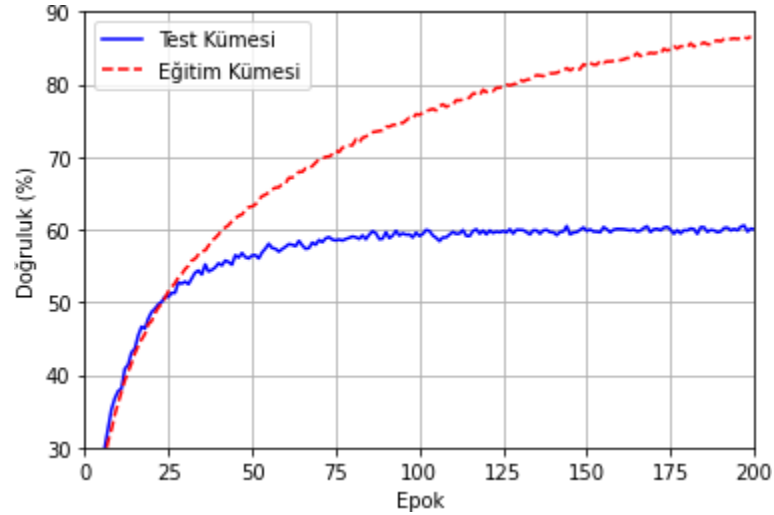
test kümesinde %45,42 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.72. VGG-19 ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.4.6. CIFAR-100 veri seti ve AlexNet-k ağ modeli ile yapılan deneyler

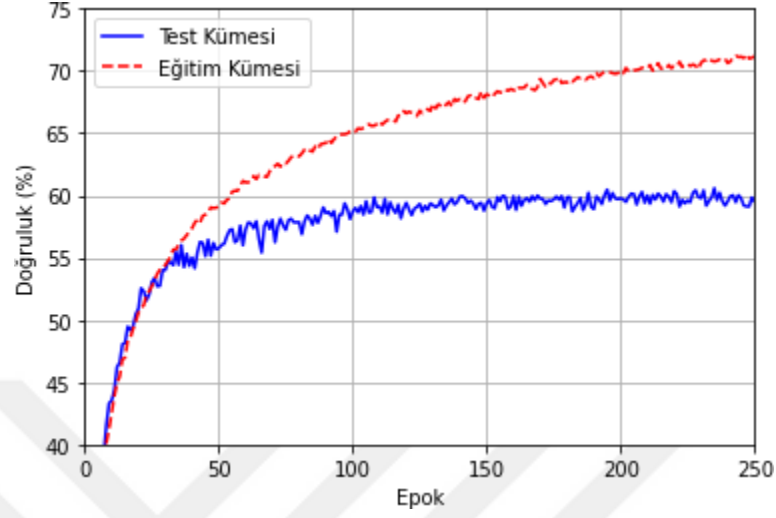
AlexNet-k ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.73'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %86,6 ve test kümesinde %60,57 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.73. AlexNet-k ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

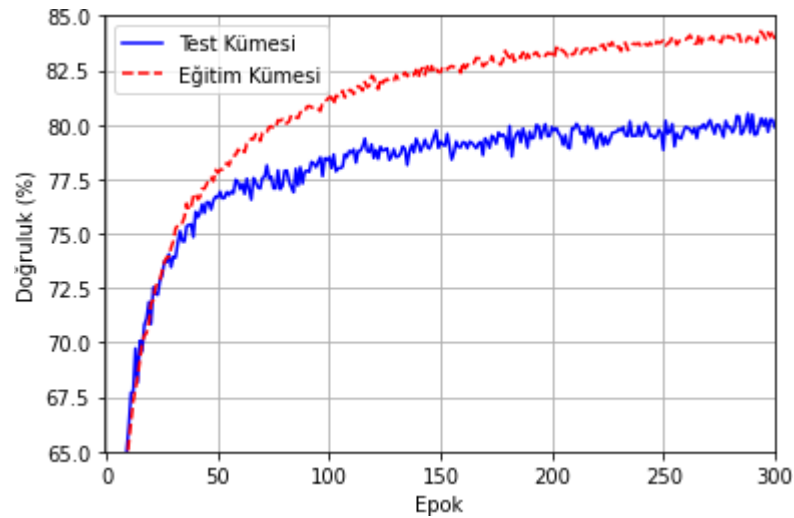
AlexNet-k ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve

eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.74'te verilmiştir. 250 epok boyunca eğitilen ağ modeli eğitim kümesinde %71,342 ve test kümesinde %60,59 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.74. AlexNet-k ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

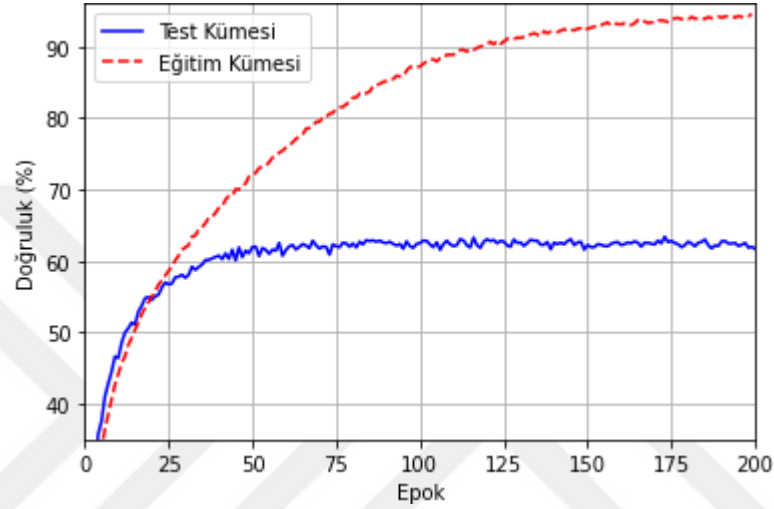
AlexNet-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.75'te verilmiştir. 250 epok boyunca eğitilen ağ modeli eğitim kümesinde %49,024 ve test kümesinde %47,2 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.75. AlexNet-k ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

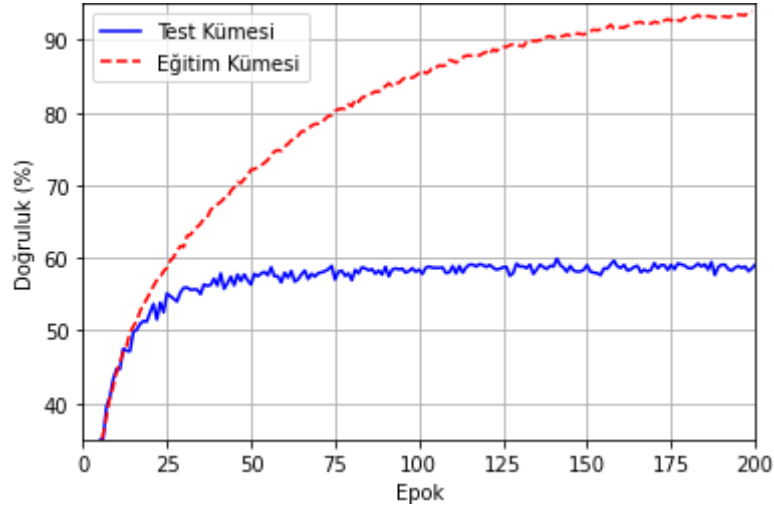
4.4.7. CIFAR-100 veri seti ve VGG-11-k ağ modeli ile yapılan deneyler

VGG-11-k ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpmaya yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.76’da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %94,568 ve test kümesinde %63,38 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



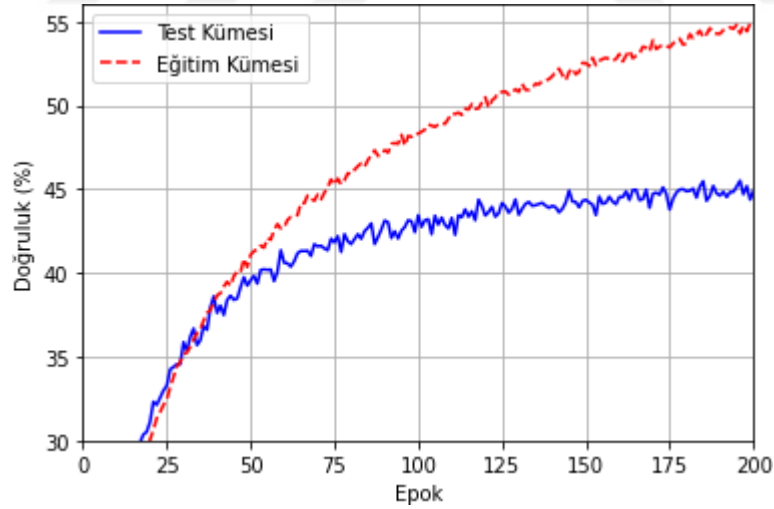
Şekil 4.76. VGG-11-k ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

VGG-11-k ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpmaya yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.77’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %93,962 ve test kümesinde %59,9 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.77. VGG-11-k ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-11-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.78’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %54,88 ve test kümesinde %45,5 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.

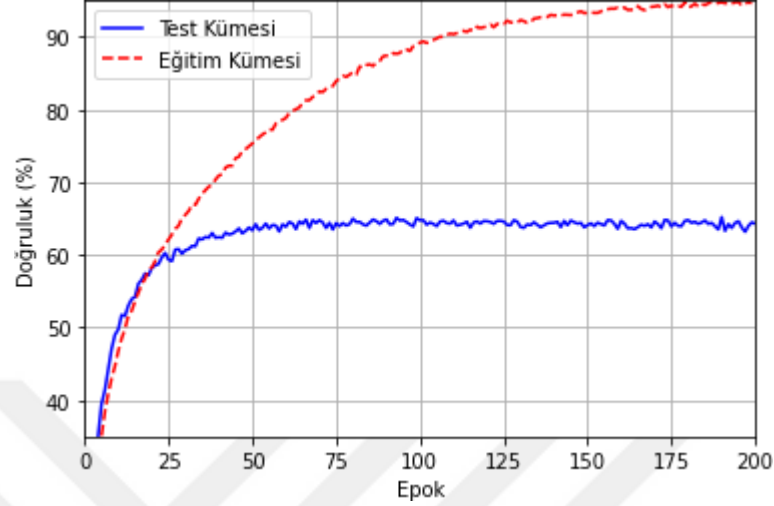


Şekil 4.78. VGG-11-k ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.4.8. CIFAR-100 veri seti ve VGG-13-k ağ modeli ile yapılan deneyler

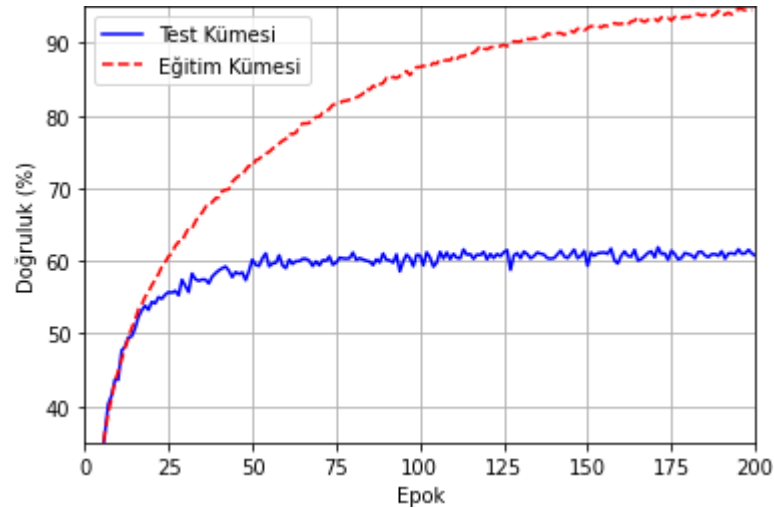
VGG-13-k ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve

eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.79'da verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %94,82 ve test kümesinde %65,21 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



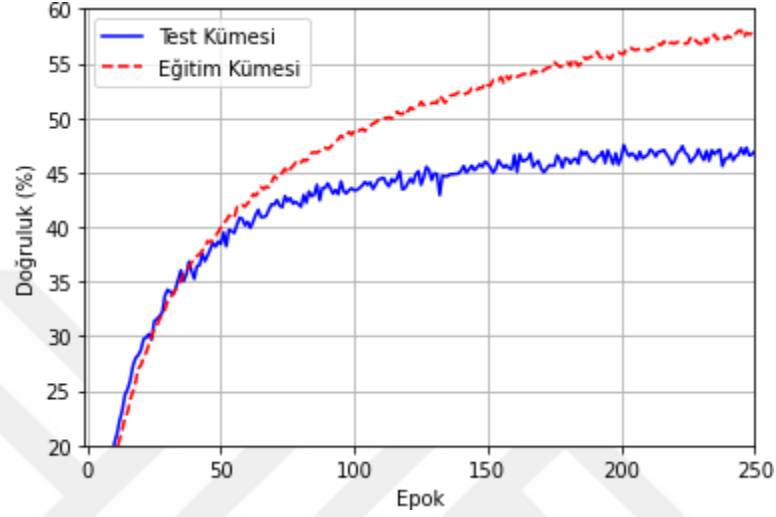
Şekil 4.79. VGG-13-k ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

VGG-13-k ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.80'de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %94,718 ve test kümesinde %61,88 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.80. VGG-13-k ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

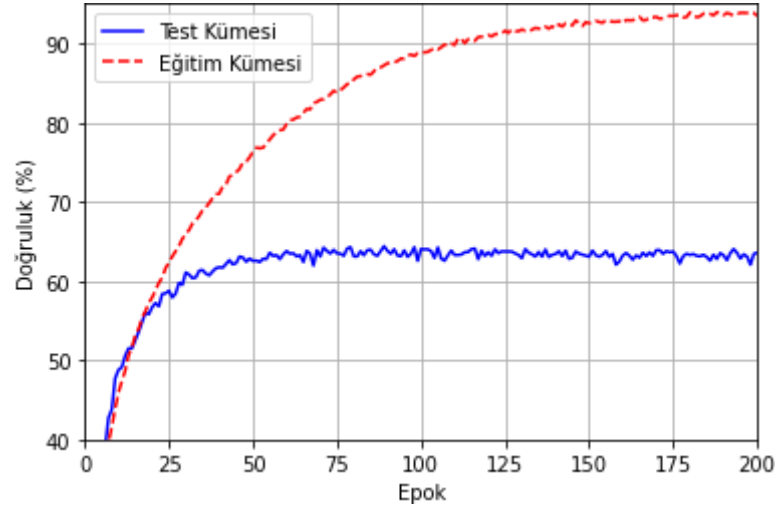
VGG-13-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.81’de verilmiştir. 250 epok boyunca eğitilen ağ modeli eğitim kümesinde %57,994 ve test kümesinde %47,52 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.81. VGG-13-k ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

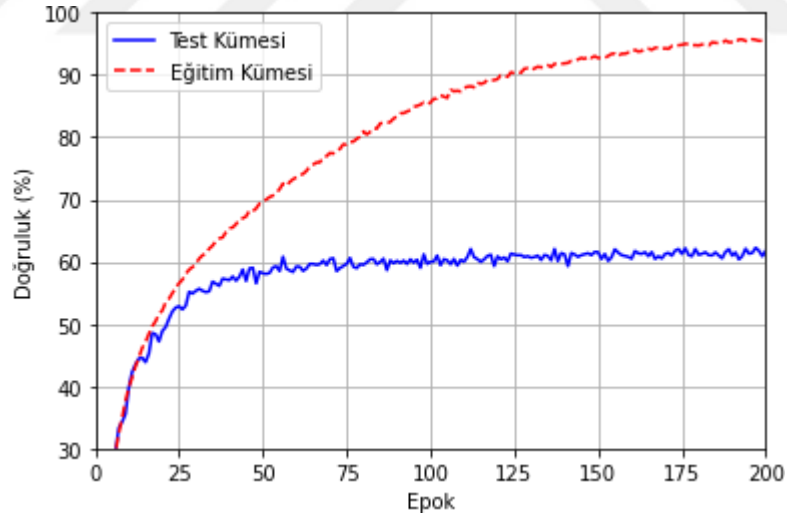
4.4.9. CIFAR-100 veri seti ve VGG-16-k ağ modeli ile yapılan deneyler

VGG-16-k ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırpma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.82’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %94,84 ve test kümesinde %64,4 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 32 olarak atanmıştır.



Şekil 4.82. VGG-16-k ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

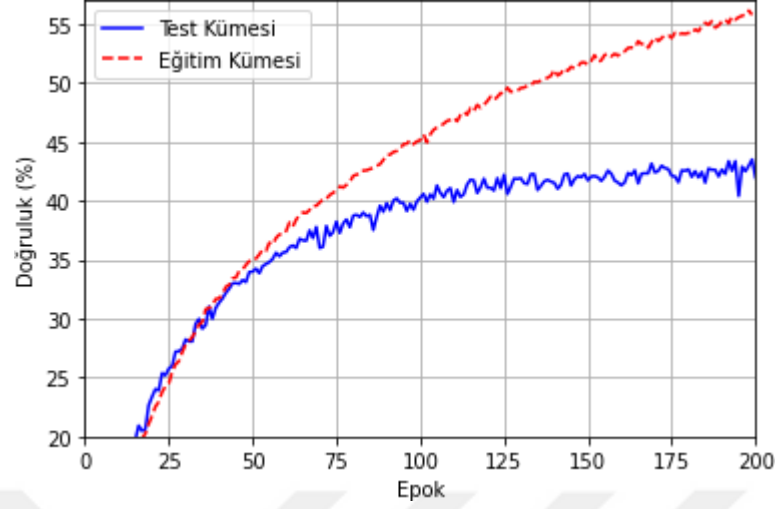
VGG-16-k ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.83'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %95,742 ve test kümesinde %62,25 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.83. VGG-16-k ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-16-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.84'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %56,16

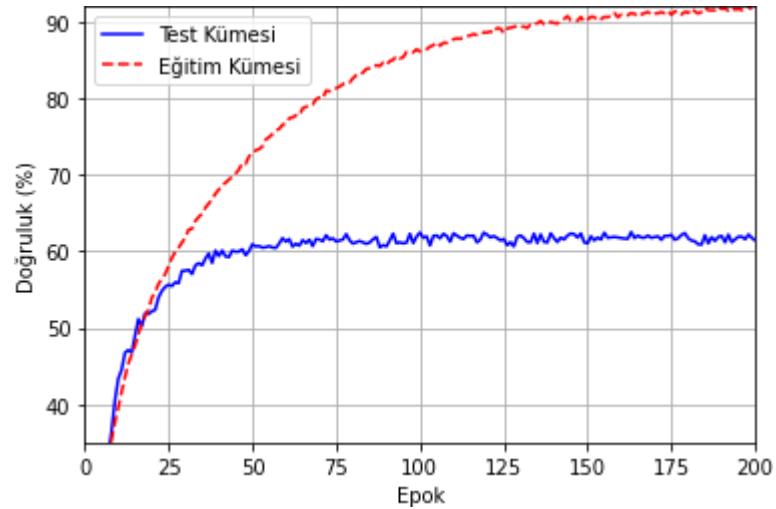
ve test kümesinde %43,52 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.84. VGG-16-k ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

4.4.10. CIFAR-100 veri seti ve VGG-19-k ağ modeli ile yapılan deneyler

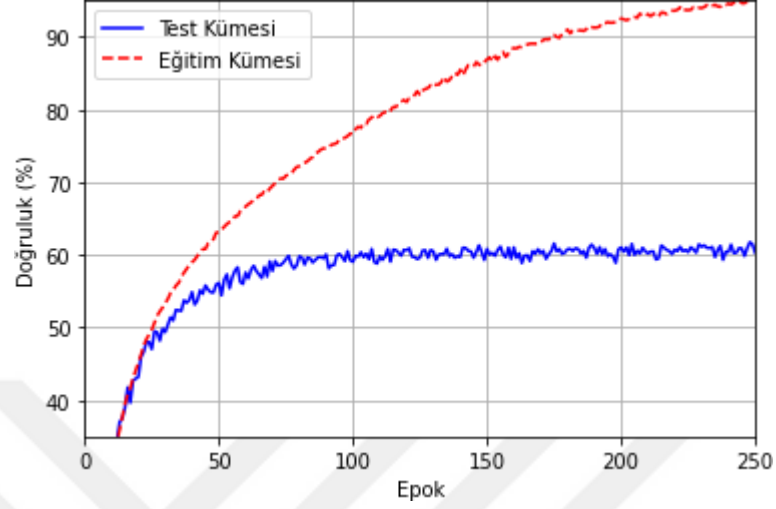
VGG-19-k ESA modeli, standart yöntemle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.85'te verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %91,986 ve test kümesinde %62,52 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.85. VGG-19-k ESA modelinin CIFAR-100 veri seti kullanılarak veri artırma yöntemi ile eğitimi

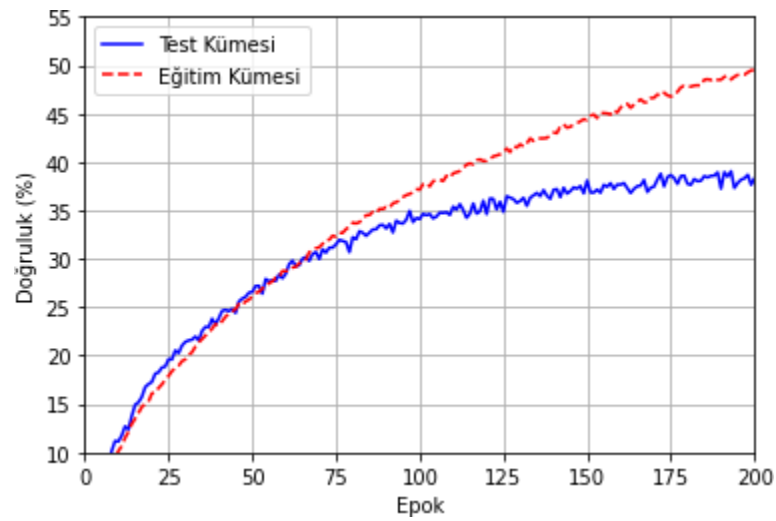
VGG-19-k ESA modeli, İAA yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve

eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.86’da verilmiştir. 250 epok boyunca eğitilen ağ modeli eğitim kümesinde %95,07 ve test kümesinde %61,76 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.86. VGG-19-k ESA modelinin CIFAR-100 veri seti kullanılarak İAA ve veri artırma yöntemi ile eğitimi

VGG-19-k ESA modeli, XNOR-Ağ yöntemiyle CIFAR-100 veri seti ve veri artırma yöntemlerinden çevirme ve rastgele kırma yöntemleri kullanılarak eğitilmiştir. Test ve eğitim veri kümeleri için doğruluk değerlerinin döneme (epok) göre değişimi Şekil 4.87’de verilmiştir. 200 epok boyunca eğitilen ağ modeli eğitim kümesinde %49,50 ve test kümesinde %39,04 sınıflandırma doğruluğuna ulaşmıştır. Yığın sayısı değeri 128 olarak atanmıştır.



Şekil 4.87. VGG-16-k ESA modelinin CIFAR-100 veri seti kullanılarak XNOR-Ağ ve veri artırma yöntemi ile eğitimi

5. SONUÇLAR VE ÖNERİLER

Bu tez çalışmasında yapılmış olan 87 adet deneye ait veriler grafiklerle desteklenerek Bölüm 4 başlığı altında sunulmuştur. Bu incelemeler sonucunda ortaya çıkan sonuçlar ve öneriler bu başlık altında verilmiştir.

5.1. Veri Artırma Yapılmayan Deneylere Ait Sonuçlar

LeNet-5 ESA modeli ile veri çoğaltma yöntemi uygulanmadan yapılan deneylere ait sonuçlar, Çizelge 5.1’de verilmiştir. LeNet-5 modeli MNIST veri setinde oldukça başarılı sonuçlar vermiştir. Sınıflandırma doğruluğu değerleri standart yöntemle SVHN veri setinde %90’ı geçerken, CIFAR-10 veri setinde ise bu değer %70’in altında kalmıştır.

Çizelge 5.1. LeNet-5 ESA modelinde sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi			Eğitim Kümesi		
	MNIST	SVHN	CIFAR-10	MNIST	SVHN	CIFAR-10
Standart	99,35	90,677	69,02	99,655	90,591	72,252
İAA	99,27	88,96	66,56	99,507	91,611	72,85
XNOR-AĞ	98,38	83,102	56,94	98,15	84,698	58,018

Çizelge 5.2’de ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (LeNet-5 modeli, veri artırma: yok) incelendiğinde ise iki yönteminde MNIST veri setinde çok az kayba uğradığı gözükmemektedir. Bu 3 veri setinde ikilileştirme yöntemlerinin en az başarılı olduğu veri seti CIFAR-10 olmuştur.

Çizelge 5.2. LeNet-5 ESA modelinde ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi			Eğitim Kümesi		
	MNIST	SVHN	CIFAR-10	MNIST	SVHN	CIFAR-10
İAA	0,081	1,894	3,564	0,149	-1,126	-0,828
XNOR-AĞ	0,976	8,354	17,502	1,510	6,505	19,700

VGG-11 ESA modeli ile veri çoğaltma yöntemi uygulanmadan yapılan deneylere ait sonuçlar, Çizelge 5.3. üzerinde verilmiştir. VGG-11 modeli, yüksek parametre ve katman sayısı ile LeNet-5 modelinden çok daha büyük ve daha derin bir ağ modelidir. VGG-11 modeli tüm veri setlerinde LeNet-5 modelinden daha iyi sonuçlar vermiştir. LeNet-5 modeli ile karşılaştırıldığında sınıflandırma doğruluğu değerleri standart yöntemle SVHN veri setinde yaklaşık %3 artarken, CIFAR-10 veri setinde ise %14 gibi önemli bir artış olmuştur.

Çizelge 5.3. VGG-11 (veri artırma: yok) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi			Eğitim Kümesi		
	MNIST	SVHN	CIFAR-10	MNIST	SVHN	CIFAR-10
Standart	99,56	93,685	83,04	99,712	98,785	98,658
İAA	99,54	94,42	83,5	100	99,997	99,756
XNOR-Ağ	99,27	92,029	77,06	99,973	99,962	99,758

Çizelge 5.4’te ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (VGG-11 modeli, veri artırma: yok) incelendiğinde ise iki yönteminde başarılarının LeNet-5 modeline göre arttığı görülmektedir. Yine bu 3 veri setinde ikilileştirme yöntemlerinin en az başarılı olduğu veri seti CIFAR-10 olmuştur. MNIST ve SVHN veri setlerinde ise standart yöntem sınıflandırma doğruluğuna oldukça yaklaşılmıştır. VGG-11 ağ modeli üzerinde SVHN ve CIFAR-10 veri setinde İAA yöntemi ve XNOR-Ağ yöntemi sınıflandırma doğrulukları LeNet-5 ağına ait sınıflandırma doğruluğundan yüksek olmuştur.

Çizelge 5.4. VGG-11 ESA modelinde (veri artırma: yok) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi			Eğitim Kümesi		
	MNIST	SVHN	CIFAR-10	MNIST	SVHN	CIFAR-10
İAA	0,020	-0,785	-0,554	-0,289	-1,227	-1,113
XNOR-Ağ	0,291	1,768	7,201	-0,262	-1,191	-1,115

VGG-19-k ESA modeli ile veri çoğaltma yöntemi uygulanmadan yapılan deneylere ait sonuçlar, Çizelge 5.5. üzerinde verilmiştir. VGG-11 modeli, yüksek parametre ve katman sayısıyla VGG-19-k modelinden çok daha fazla parametreye sahip bir modeldir. Buna rağmen 3 veri setinde de standart ve İAA yöntemlerinde VGG-19-k ağ modeli daha iyi sonuçlar vermiştir.

Çizelge 5.5. VGG-19-k (veri artırma: yok) Sınıflandırma doğrulukları (%)

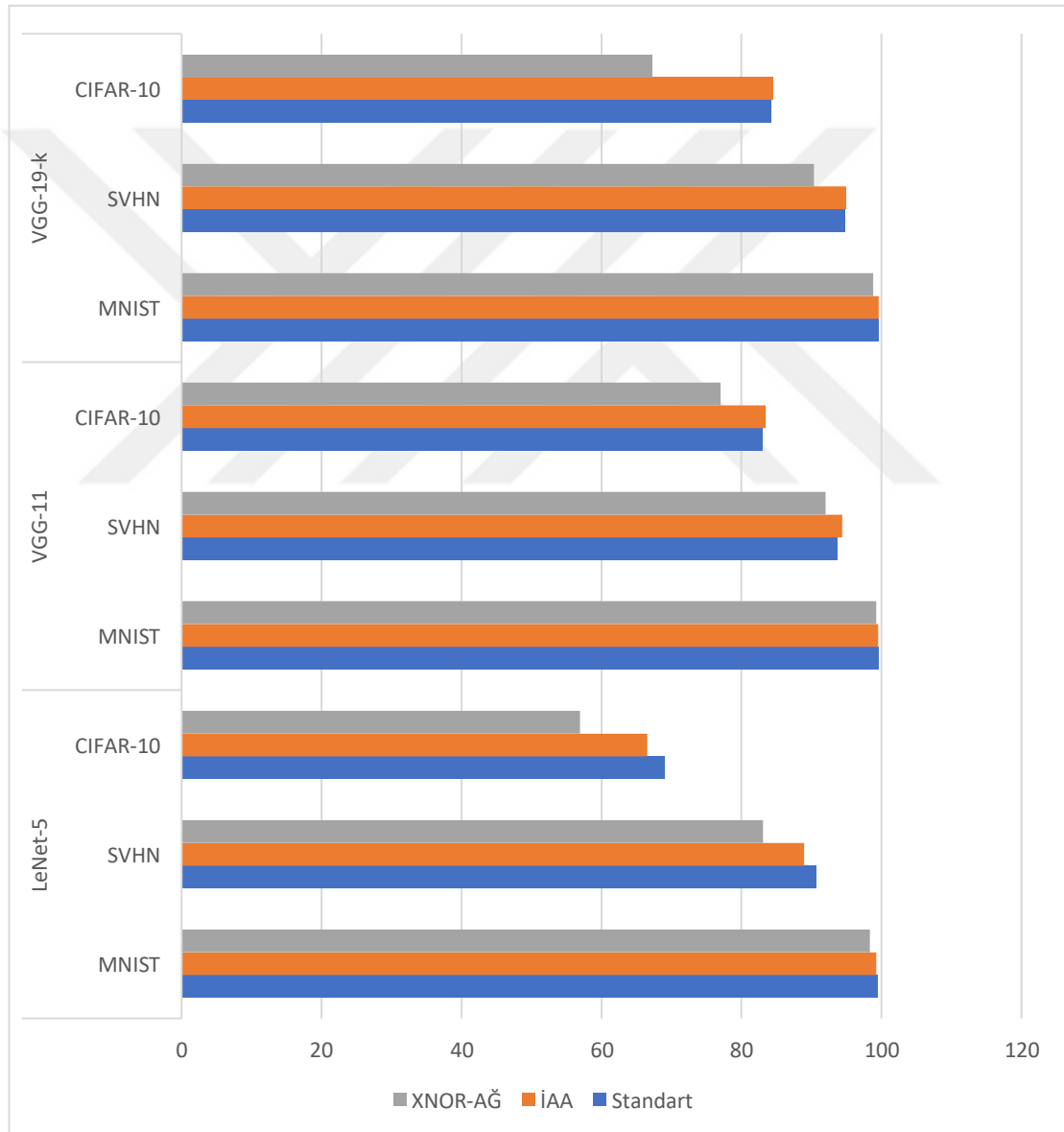
Yöntem	Test Kümesi			Eğitim Kümesi		
	MNIST	SVHN	CIFAR-10	MNIST	SVHN	CIFAR-10
Standart	99,63	94,707	84,18	99,985	99,749	99,65
İAA	99,62	94,975	84,49	100	99,881	99,832
XNOR-Ağ	98,85	90,362	67,23	99,627	98,11	93,302

Çizelge 5.6’da ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (VGG-19-k modeli, veri artırma: yok) incelendiğinde VGG-19-k modelinde XNOR-Ağ yönteminde kayıp oranı VGG-11 modelinden daha yüksek olmuştur.

Çizelge 5.6. VGG-19-k ESA modelinde (veri artırma: yok) ikileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi			Eğitim Kümesi		
	MNIST	SVHN	CIFAR-10	MNIST	SVHN	CIFAR-10
İAA	0,010	-0,283	-0,368	-0,015	-0,132	-0,183
XNOR-AĞ	0,783	4,588	20,135	0,358	1,643	6,370

VGG-19-k, VGG-11 ve LeNet-5 modellerinde veri artırma yöntemi kullanılmadan yapılan çalışmalarda test veri kümelerine ait sınıflandırma doğruluğu değerleri grafiği Şekil 5.1’de verilmiştir.



Şekil 5.1. Veri artırma yapılmayan çalışmalara ait sınıflandırma doğrulukları (%)

5.2. Veri Artırma Yapılan Deneylere Ait Sonuçlar

AlexNet ESA modelinin CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu değerleri Çizelge 5.7’de verilmiştir. Çizelge 5.8’de ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

Çizelge 5.7. AlexNet (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	88,35	64,43	99,53	98,916
İAA	88,45	64,73	98,92	94,758
XNOR-AĞ	83,39	55,65	94,364	72,812

Çizelge 5.8. AlexNet ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	-0,113	-0,466	0,613	4,204
XNOR-AĞ	5,614	13,627	5,190	26,390

VGG-11 ESA modelinin CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu değerleri Çizelge 5.9’da verilmiştir. Çizelge 5.10’da ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

Çizelge 5.9. VGG-11 (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	89,94	66,14	98,458	96,338
İAA	88,82	64,52	99,468	99,636
XNOR-AĞ	82,1	52,03	95,312	81,042

Çizelge 5.10. VGG-11 ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	1,245	2,449	-1,026	-3,423
XNOR-AĞ	8,717	21,334	3,195	15,877

VGG-13 ESA CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu

değerleri Çizelge 5.11’de verilmiştir. Çizelge 5.12.’de ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

Çizelge 5.11. VGG-13 (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	90,59	67,71	99,592	99,784
İAA	91,01	67,29	99,496	99,774
XNOR-AĞ	84,1	54,7	95,908	82,764

Çizelge 5.12. VGG-13 ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	-0,464	0,620	0,096	0,010
XNOR-AĞ	7,164	19,214	3,699	17,057

VGG-16 ESA modelinin CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu değerleri çizelge 5.13.’da verilmiştir. Çizelge 5.14.’te ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

Çizelge 5.13. VGG-16 (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	91,93	68,32	99,056	96,318
İAA	91,22	68,34	99,354	99,418
XNOR-AĞ	82,82	49,95	95,548	82,834

Çizelge 5.14. VGG-16 ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	0,772	-0,029	-0,301	-3,219
XNOR-AĞ	9,910	26,888	3,541	13,999

VGG-19 ESA modelinin CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu değerleri Çizelge 5.15’te verilmiştir. Çizelge 5.16’da ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

Çizelge 5.15. VGG-19 (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	90,71	65,59	99,592	99,634
İAA	91,32	66,57	99,182	97,294
XNOR-AĞ	80,03	45,42	92,732	74,244

Çizelge 5.16. VGG-19 ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	-0,672	-1,494	0,412	2,349
XNOR-AĞ	11,774	30,752	6,888	25,483

AlexNet-k ESA modelinin CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu değerleri Çizelge 5.17’de verilmiştir. Çizelge 5.18’de ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

Çizelge 5.17. AlexNet-k (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	87,32	60,57	98,482	86,6
İAA	86,72	60,59	94,558	71,342
XNOR-AĞ	80,51	47,2	84,28	49,024

Çizelge 5.18. AlexNet-k ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	0,687	-0,033	3,984	17,619
XNOR-AĞ	7,799	22,074	14,421	43,390

VGG-11-k ESA modelinin CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu değerleri Çizelge 5.19’da verilmiştir. Çizelge 5.20’de ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

Çizelge 5.19. VGG-11-k (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	88,81	63,38	97,166	94,568
İAA	86,15	59,9	98,294	93,962
XNOR-AĞ	77,78	45,5	85,188	54,88

Çizelge 5.20. VGG-11-k ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	2,995	5,491	-1,161	0,641
XNOR-AĞ	12,420	28,211	12,327	41,968

VGG-13-k ESA modelinin CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu değerleri Çizelge 5.21’de verilmiştir. Çizelge 5.22’de ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

Çizelge 5.21. VGG-13-k (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	90,35	65,21	97,944	94,82
İAA	88,4	61,88	98,354	94,718
XNOR-AĞ	79,69	47,52	86,236	57,994

Çizelge 5.22. VGG-13-k ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	2,158	5,107	-0,419	0,108
XNOR-AĞ	11,799	27,128	11,954	38,838

VGG-16-k ESA modelinin CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu değerleri Çizelge 5.23’te verilmiştir. Çizelge 5.24’te ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

Çizelge 5.23. VGG-16-k (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	90,37	64,4	98,072	94,84
İAA	88,92	62,25	97,368	95,742
XNOR-AĞ	77,79	43,52	85,796	56,16

Çizelge 5.24. VGG-16-k ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	1,605	3,339	0,718	-0,951
XNOR-AĞ	13,921	32,422	12,517	40,784

VGG-19-k ESA modelinin CIFAR-10 ve CIFAR-100 veri setleri üzerinde veri artırma uygulanarak standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu değerleri Çizelge 5.25'te verilmiştir. Çizelge 5.26'da ikilileştirme yöntemlerinin standart yönteme göre yüzdesel kayıp oranları verilmiştir.

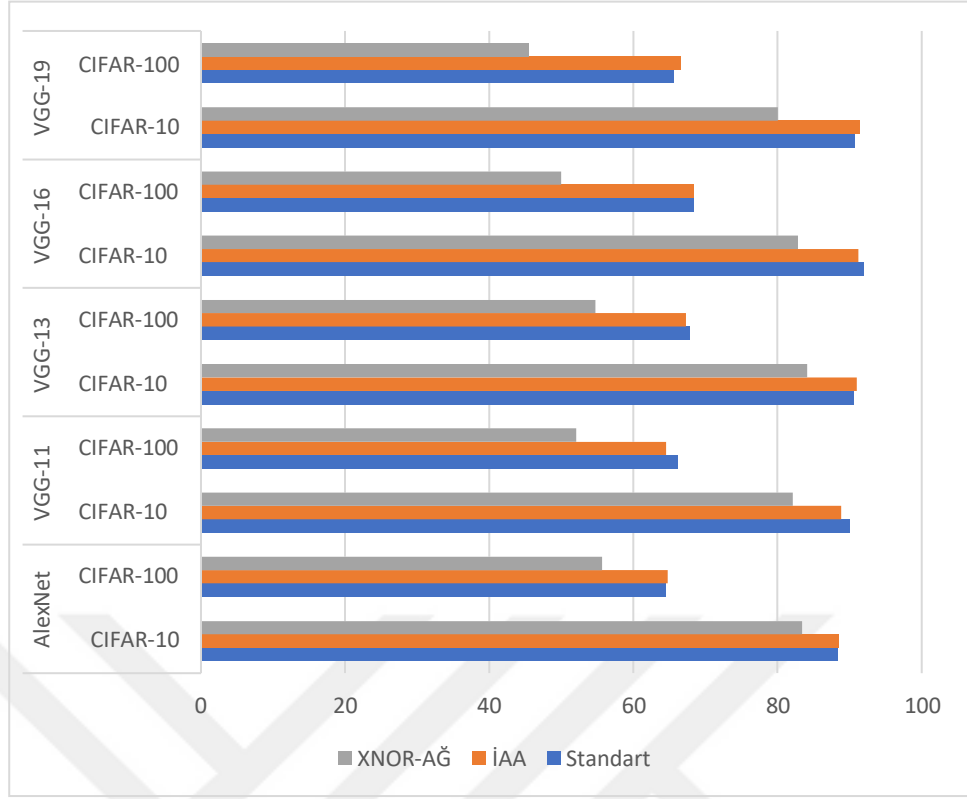
Çizelge 5.25. VGG-19-k (veri artırma: var) Sınıflandırma doğrulukları (%)

Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standart	90,43	62,52	97,994	91,986
İAA	89,02	61,76	97,068	95,07
XNOR-AĞ	71,96	39,04	78,066	49,506

Çizelge 5.26. VGG-19-k ESA modelinde (veri artırma: var) ikilileştirme yöntemlerinin standart yönteme göre kayıp oranları (%)

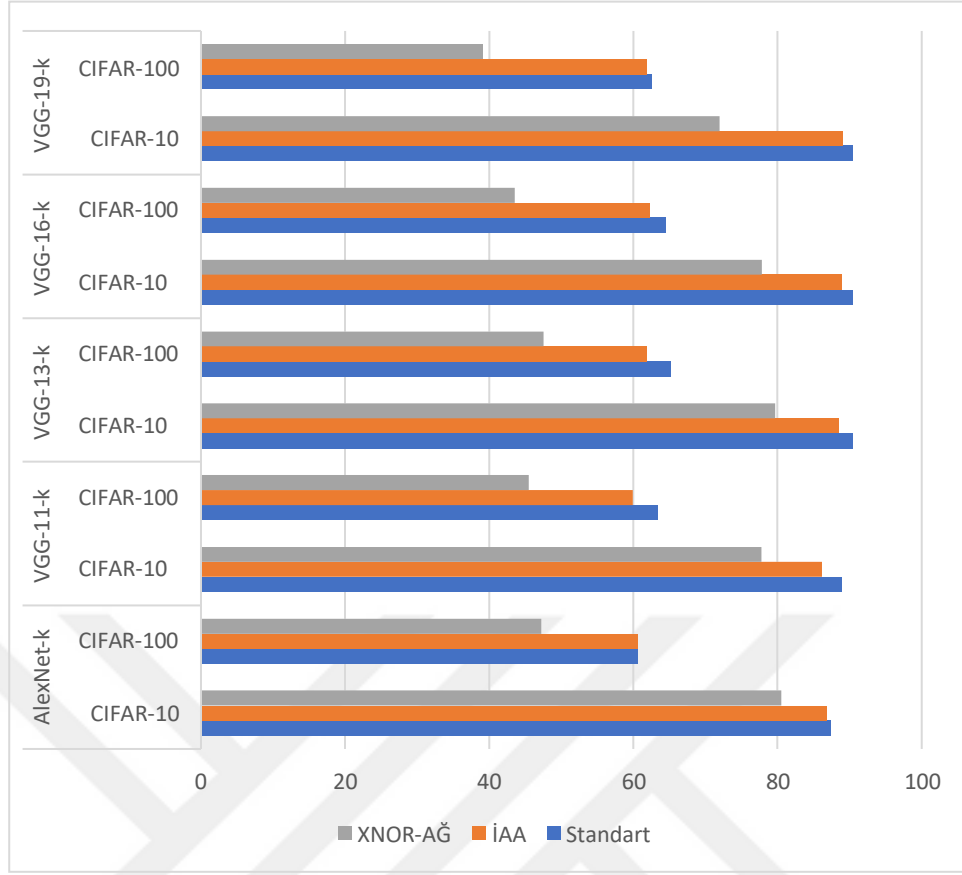
Yöntem	Test Kümesi		Eğitim Kümesi	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
İAA	1,559	1,216	0,945	-3,353
XNOR-AĞ	20,425	37,556	20,336	46,181

AlexNet ve VGG ağ modelleri kullanılarak yapılan çalışmalarda standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu (test kümesi) grafiği Şekil 5.2'de verilmiştir.



Şekil 5.2. AlexNet ve VGG ağ modellerine ait sınıflandırma doğruluğu değerleri

Bu tez çalışması için tasarlanan ağ modelleri kullanılarak yapılan çalışmalarda standart, İAA ve XNOR-Ağ yöntemlerine ait sınıflandırma doğruluğu (test kümesi) grafiği Şekil 5.3'te verilmiştir.



Şekil 5.3. Özelleştirilmiş ağ modellerine ait sınıflandırma doğruluğu değerleri

Bu tez çalışmasında yapılan deneylerde Standart yönteme, İAA yöntemine ve XNOR-Ağ yöntemine ait sınıflandırma doğruluk değerlerinin ortalamaları Çizelge 5.27’de verilmiştir.

Çizelge 5.27. Bu tez çalışmasında yapılan verilere ait ortalamalar

Yöntem	Sınıflandırma Doğruluğu Ortalamaları (%)
Standart	81,411
İAA	80,661
XNOR-Ağ	70,48

Bu tez çalışması için özelleştirilmiş ve parametre sayıları azaltılmış modeller ile standart modeller

Bu tez çalışmasında yapılan deneylerden alınan verilerle, ESA’larda standart yönteme göre İAA yönteminin ve XNOR-Ağ yönteminin kazanç ve kayıplarını gösteren karşılaştırma Çizelge 5.29’da verilmiştir (Govindu ve ark 2004, Beauchamp ve ark 2006, Horowitz 2014, Rastegari ve ark 2016). İAA yönteminde ortalama sınıflandırma doğruluğu kaybı %1,019 olmuştur. XNOR-Ağ yönteminde ortalama sınıflandırma doğruluğu kaybı %14,839 olmuştur.

Çizelge 5.29. İkileştirme yöntemlerine ait kayıp ve kazanç oranları

Yöntem	Ortalama Kayıp	CPU Hız kazancı	Bellek Kazancı	Enerji Tasarrufu
İAA	%1,019	2x	32x/64x	>32x
XNOR-Ağ	%14,839	58x	32x/64x	>32x

Bellek kazancı: Tek duyarlıklık gösterim / Çift duyarlıklık gösterim

5.3. Öneriler

Bu tez çalışmasında yapılan çalışmalar göstermiştir ki ağda parametre alan katman sayısı arttıkça İAA yönteminin sınıflandırma başarısı standart yapıya daha da yaklaşabilmektedir. Aynı zamanda katmanlarda bulunan parametre sayılarındaki artış da İAA yönteminin sınıflandırma doğruluğu başarısını artırmaktadır. Bu verilere göre İAA yöntemi kullanılacağında parametre alan katman sayısının ve katmanlardaki parametre sayısının yüksek tutulması önerilebilir.

Bu tez çalışmasında yapılan çalışmalar göstermiştir ki XNOR-Ağ yönteminde 13 kadar parametre alan katmana sahip ağlarda sınıflandırma başarısında artış görülürken 16 ve 19 katmanlı ağlarda yöntemin başarısı düşmektedir. Bu yüzden XNOR-Ağ yöntemi kullanılacak ağ modellerinde 13 ve daha az katmanlı modeller kullanılması önerilebilir.

Bu tez çalışmasında yapılan çalışmalarda katmanlarda bulunan parametre sayılarındaki artışın XNOR-Ağ yönteminin sınıflandırma doğruluğu başarısını artırdığı görülmektedir. Bu verilere göre XNOR-Ağ yöntemi kullanılacağında katmanlardaki parametre sayısının yüksek tutulması önerilebilir.

Taşınabilir aygıtlar gibi donanım kısıtı olan sistemlerde VGG ağları gibi büyük ağlar kullanılmadığında LeNet-5 gibi küçük ağ yapıları tercih edilmektedir. Küçük ESA modellerinin kullanılması yerine İAA yöntemi ya da XNOR-Ağ yöntemi kullanılarak VGG-11 gibi yüksek işlem ve bellek maliyetine sahip olan ağların ikileştirilerek kullanılması önerilebilir. Bu çalışmada VGG-11 ve VGG-19-k ESA modellerinde ikileştirme yöntemlerinin sınıflandırma doğruluğu değerleri hem CIFAR-10 hem de SVHN veri setinde LeNet-5 standart modelini geçmiştir. Bu durum öneriyi desteklemektedir.

Ağa giriş olarak verilen görüntü 32x32 boyutlarında ise bu tez çalışması için AlexNet ve VGG modellerini özelleştirerek tasarlanan daha az parametrelili modeller kullanılabilir.

Bellek kısıtı olan sistemlerde makul oranda sınıflandırma kaybı göze alınarak İAA yöntemi kullanılabilir. Hem bellekten 32 kat kazanç hem de 2 kat işlem hızı kazancı sağlanabilir. Bellek donanımları maddi olarak pahalı aygıtlardır. Bu sayede sistem maliyeti düşürülebilir.

XNOR-Ağ yöntemi kullanılarak tasarlanan ağ modelleri GPU yerine CPU'lar üzerinde çalışabildiği için sınıflandırma doğruluğundan ödün verilerek daha hafif, küçük ve ucuz sistemler tasarlanabilir.

Günümüz taşınabilir sistemlerinde en büyük kısıtlardan biri de pil ömrü olmaktadır. Aynı zamanda taşınabilir olmayan sistemler için de enerji maliyeti problem olabilmektedir. Bu sorunu çözmek için İAA ve XNOR-Ağ yöntemleri kullanılabilir.

Gri format görüntülerin sınıflandırıldığı uygulamalarda ağ ikilileştirme yöntemlerinin denenmesi fayda sağlayabilir.

Veri artırma yönteminin ikilileştirilmiş ağları eğilirken kullanılması da sınıflandırma başarısını artırabilmektedir.

Gerçek zamanlı uygulamalarda ESA'lar, işleyen sistemin üzerinde değil uzaktan bir bağlantı ile başka bir yerde çalışıp sonuçları yine uzaktan bağlantı ile göndermekte olabiliyorlar. Bu şekilde çalışan sistemlerde senkronizasyon bozuklukları meydana gelebilmektedir. Özellikle askeri sistemler gibi hayati önem arz eden uygulamalarda senkronizasyon bozuklukları ve bağlantı sorunları ciddi problemler oluşturur. Bu gibi uygulamalar da sınıflandırma başarısından ödün verilerek XNOR-Ağ yöntemi ana sınıflandırıcıyı tasarlamak için kullanılabilir. Ya da bu gibi önem arz eden uygulamalar da XNOR-Ağ yöntemi ile tasarlanan ağ yeterli ve nispeten düşük bir donanımla işleyen sistemin üzerinde yedek olarak bulundurulabilir. Uzaktan bağlantı ile ilgili sorunlar oluştuğunda devreye girebilir. Bu sayede sistemin çalışması kesintiye uğramadan devam edebilir.

Bu ikilileştirme yöntemleri, azalan eğim temelli geri beslemeli tüm derin sinir ağlarına teorik olarak adapte edilebilir. Bu sayede bellek, işlem kazancı ve enerji tasarrufu sağlanabilir.

KAYNAKLAR

- Amidi A , Amidi S. 2018. CS 230-Convolutional Neural Networks Cheatsheet. URL: <https://stanford.edu/shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks>.
- Beauchamp MJ, Hauck S, Underwood KD , Hemmert KS. 2006. Embedded floating-point units in FPGAs. Proceedings of the 2006 ACM/SIGDA 14th international symposium on Field programmable gate arrays. 12-20.
- Courbariaux M, Bengio Y , David J-P. 2015. Binaryconnect: Training deep neural networks with binary weights during propagations. Advances in neural information processing systems. 3123-31.
- Goodfellow I, Bengio Y , Courville A. 2016. Deep learning: MIT press.
- Govindu G, Zhuo L, Choi S , Prasanna V. 2004. Analysis of high-performance floating-point arithmetic on FPGAs. 18th International Parallel and Distributed Processing Symposium, 2004. Proceedings. IEEE, 149.
- He K, Zhang X, Ren S , Sun J. 2016. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition. 770-8.
- Hubara I, Courbariaux M, Soudry D, El-Yaniv R , Bengio Y. 2016. Binarized neural networks. Advances in neural information processing systems. 4107-15.
- Ioffe S , Szegedy C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.
- Krizhevsky A , Hinton G. 2009. Learning multiple layers of features from tiny images.
- Krizhevsky A, Sutskever I , Hinton GE. 2012. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems. 1097-105.
- Lecun Y, Bengio Y , Hinton G. 2015. Deep learning. nature 521;436-44.
- Lecun Y, Bottou L, Bengio Y , Haffner P. 1998a. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86;2278-324.
- Lecun Y, Bottou L, Bengio Y , Haffner P. 1998b. Gradient-based learning applied to document recognition. 86;2278-324.
- Netzer Y, Wang T, Coates A, Bissacco A, Wu B , Ng AY. 2011. Reading digits in natural images with unsupervised feature learning.
- Özkan İ , Ülker E. 2017a. Derin Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri. Gaziosmanpaşa Bilimsel Araştırma Dergisi 6;85-104.
- Rastegari M, Ordonez V, Redmon J , Farhadi A. 2016. Xnor-net: Imagenet classification using binary convolutional neural networks. European conference on computer vision. Springer, 525-42.
- Simonyan K , Zisserman A. 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
- Skansi S. 2018. Introduction to Deep Learning: from logical calculus to artificial intelligence: Springer.
- Srivastava N, Hinton G, Krizhevsky A, Sutskever I , Salakhutdinov R. 2014. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research 15;1929-58.
- Sun X, Yin S, Peng X, Liu R, Seo J-S , Yu S. 2018. XNOR-RRAM: A scalable and parallel resistive synaptic architecture for binary neural networks. 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE). IEEE, 1423-8.

- Szegedy C, Liu W, Jia Y, et al. 2015. Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition. 1-9.
- Yacim JA , Boshoff DGB. 2018. Impact of artificial neural networks training algorithms on accurate prediction of property values. Journal of Real Estate Research 40;375-418.
- Zeiler MD , Fergus R. 2014. Visualizing and understanding convolutional networks. European conference on computer vision. Springer, 818-33.



