CIFAR-10 Veri Seti ile Farklı Sınıflandırıcıların Performans Karşılaştırması

Safa Demirhan   
Bilgisayar Mühendisliği  
TOBB ETÜAnkara, Türkiye  
s.demirhan@etu.edu.tr

*Özet*—Bu çalışmanın amacı CIFAR-10 veri seti kullanılarak eğitilen modellerin ne derecede iyi çalışacaklarını gözlemlemek ve bu modelleri birbirleri ile karşılaştırmaktır.

Anahtar Kelimeler—CIFAR-10, CNN, SVM, Rastgele Orman, Logistic Regression, Naïve Bayes, MLP, Bilgisayarda görü

# **GİRİŞ**

CIFAR-10 veri seti, bilgisayarda görüye girişte sık sık kullanılan veri setlerindendir. Veri, 32x32 büyüklüğündeki 60.000 görseli, ait olduğu 10 sınıftan biriyle eşleştirmektir. Bu sınıflar, uçak, araba, kuş, geyik, köpek, kurbağa, at, gemi ve kamyondan oluşmaktadır. Eşleştirmeyi yaparken çeşitli modeller kullanılmıştır ve bu modeller doğruluk oranları göz önünde bulundurularak birbirleri ile karşılaştırılmıştır. Eğitilen modeller arasında CNN, SVM, k en yakın komşu, rastgele orman, lojistik regresyon ve Naïve Bayes modelleri yer almaktadır. Bu farklı modellerin karşılaştırmak için kullanılacak başarım metrikleri, doğruluk oranı, kesinlik (ing. Precision) ve anımsamadır (ing. Recall).

# **LİTERATÜR ARAŞTIRMASI**

Literatürde, çok araştırılan ve giriş seviyesinde bir problem olması sebebiyle birçok kaynak yer almaktaydı. CIFAR-10 için özel geliştirilmiş modeller, araştırma boyunca en yüksek doğruluk oranını veren sınıflandırıcılar oldular. Bunlar karışık ve büyük modeller olduğundan eğitilen modeller arasında yer almamaktadırlar. Bu araştırma kapsamınca, en bilindik sınıflandırıcıları kullanarak CIFAR-10 problemine çözüm aranmıştır. Yine de literatürde bu problem üzerine en iyi sınıflandırma hangi modeller ile sağlanmıştır görelim:

|  |  |  |
| --- | --- | --- |
| Model | Doğruluk (%) | Yıl |
| ViT-H/14 | 99.5 | 2020 |
| ViT-L/16 | 99.42 | 2020 |
| CaiT-M-36 U 224 | 99.4 | 2021 |
| BiT-L (ResNet) | 99.37 | 2019 |
| EfficientNetV2-L | 99.1 | 2021 |

Veri setini sınıflandırırken genellikle CNN’e başvurulmaktadır. Bu bağlamda CNN ile ortalama %18’lik test verisi hatası yapılmaktadır, veri artırma (ing. Data augmentation) bu oran %11’lere kadar düşer. Bayesian hiperparametre optimizasyonu kullanarak, veri çoğaltmasız %15’lik bir test seti hata oranı elde edilebilir. CIFAR-10 adı, Canadian Institute For Advanced Research’ten gelmektedir. Bu çoğunluğunun Kanada’da ikamet ettiği araştırmacı grubu, sağlık, sosyalizm, madde, dünya ve uzay gibi çeşitli konularda araştırmalar yürütmektedir.

# **VERİ SETİ, VERİ ÖZELLİKLERİ, ÖZNİTELİKLER**

## -Veri Kaynağı

Veri seti, bu problemde çalışan herkesin de kullandığı ve Keras kütüphanesinde dahi yer alan üzerine çok araştırma yapılmış bir veri setidir.

Veri seti linki: [www.cs.toronto.edu/~kriz/cifar.html](http://www.cs.toronto.edu/~kriz/cifar.html)

## -Veri Kümesi

Veri tamamen görsellerden oluştuğu için öznitelikler piksel gibi düşünülebilir. Tabii ki burada 32x32 bir görsel için 1024 öznitelik düşünmek yanlış olacaktır. CNN için örnek vermek gerekirse, görseller önce 1024 uzunluğunda bir diziye dönüştürülürler ancak bu bir öznitelik kümesinde öte verinin imzasıdır. Bu dizi üzerinde bir takım işlemler yapılır. 1024 pikselin her bir değerinin sonucu git gide azalarak 10 sınıftan hangisiyle eşleşeceğine kadar iner. Bu işlemlerden bazıları Convolution, MaxPooling ve Dropout’tur.

## -Proje Metodolojisi

**Şekil 1.** Projenin işleyiş şeması

## -Önişleme Aşamaları

Önişleme olarak öncelikle veri setindeki görsellerin pikselleri [0,1] aralığına ölçeklendirildi. 10 sınıfa da one-hot encoding uygulandı. Veri ölçeklendirme, özellikle bilgisayarda görü problemlerinde modelin daha iyi öğrenebilmesi için şarttır. Verideki 60.000 verinin, ne modeli yeterince eğitemeyecek kadar, ne de yeterli test verisi bırakmayacak şekilde bölünmesi de çok önemlidir. Önişleme aşaması olarak verinin eğitim ve test verisi olmak üzere iki kümeye ayrılması da dahildir. Bunlar da kendi içerisinde kaynak ve hedef olmak üzere ikiye ayrılabilir. Sıklıkla X\_train ve y\_train şeklinde adlandırılan verilerdir.

Görüntüyü işleyebilmek için sırasıyla her pikselin değeri saklanır ve bunlar çeşitli aşamalardan geçirilerek sonuç elde edilir. Bunun için de her görsel öncelikle Dense kullanılarak iki boyuttan tek boyutlu bir veri yapısına indirgenir. Böylece görseldeki piksellerin hangi sırayla geldikleri de önemini yitirmemiş olur.

## -Öznitelik Açıklamaları

Bilgisayarda görü problemlerinde genellikle tek öznitelik piksellerdir. Bu durumda hatalı veriler göz önünde bulundurulduğunda, bu problem için sorun oluşturmamaktadır çünkü veri setinde bozulmuş bir görsel de yoktur. Aynı şekilde eksik veriler de yoktur. Veri setindeki her bir veri parçası iki boyutlu bir tam sayı dizisidir aslında. Bunun nedeni piksellerin [0,255] aralığında yer almasıdır. Dizi yapısından söz edildiğinde tabii ki de her bir pikselin sırasının önemi vardır ancak görsellerin kendi içinde bir sırası yoktur.

# **KULLANILAN MODELLER**

## CNN

Evrişimli sinir ağı (ing. Convolutional neural network) bilgisayarda görüde kullanılması kaçınılmaz bir sınıflandırıcıdır. CIFAR-10 için de, en yüksek doğruluk oranını veren sınıflandırıcı %85 ile CNN oldu. Modeli hazırlarken, Convolution2D, BatchNormalization, MaxPooling2D, Dropout, Flatten, Dense gibi katmanları birden fazla defa arka arkaya eklemenin de modelin başarımını artırdığı gözlemlendi. Bunun nedeni her eklenen katmanla modelin daha detaylı kararlar verebilmesi ve aşırı öğrenmeden (ing. Overfitting) de kaçınıyor olmasıdır. Ayrıntılı karardan kasıt, örneğin eski bir araba görseli, şeklinden ötürü bir kamyona benziyor olabilir. Yukarıda bahsi geçen sınıfları kullanarak bu küçük farklara sahip görsellerin de doğru tahmin edilmesi mümkün olabiliyor. Katmanlı yapının kaç tane olacağı konusu ise deneme yanılma ile test edilmiştir. Başarım küçük oranlarda olsa dahi artıyorsa katmanlar eklenmeye devam etmiştir.

Optimizasyon sınıfı olarak Adam tercih edilmiştir ve kayıp fonksiyonu (ing. Loss function) olarak categorical\_crossentropy seçilmiştir. Bunun nedeni 10 sınıftan hangisi olacağını sınıflandırmak hedefimiz olduğu içindir.

Veri çoğaltma, sınıflandırıcıyı daha fazla eğitirken aynı zamanda aşırı öğrenmeden de korur. Bunun nedeni eldeki veriye çeşitli işlemlerin rastgele uygulanmasıdır. Bunlardan birkaçı, görseli döndürmek, kaydırmak, renklerini değiştirmek, simetrisini almak, görseli kısmen kapatan nesneler yerleştirmek ve bu işlemleri birkaçını aynı görselde arka arkaya uygulamak.

**Şekil 2.** CNN'nin bir katmanındaki öznitelik gösteriminin t-SNE grafiği.

Erken durma (ing. Early stopping) ve öğrenme oranını

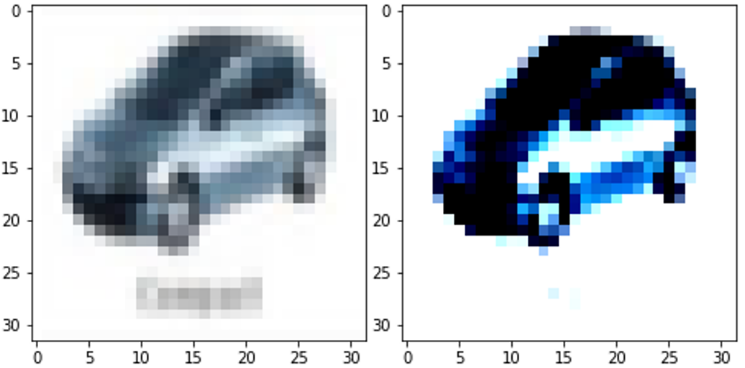
azaltma (ing. Learning rate annealer) CNN için bahsedeceğimiz son yardımına başvurulan yöntemlerdir. Bu iki yöntem genelde model büyük devrim (ing. Epoch) sayılarında eğitileceği zamanlar devreye girer. Başarımda az da olsa bir artış sağlamıştır.

## SVM

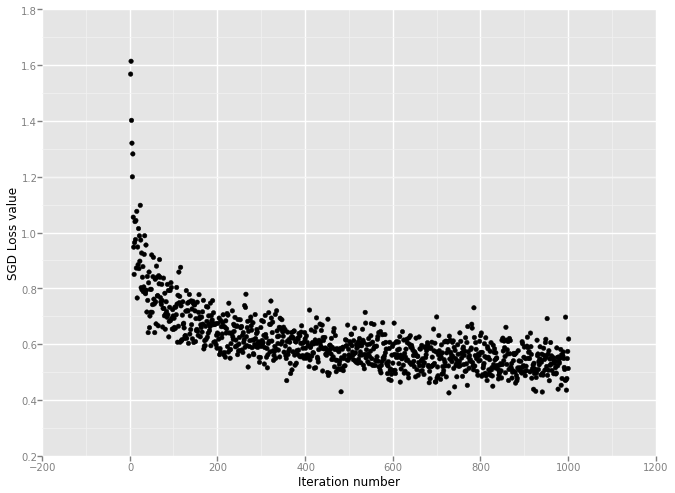
Sınıflandırma için bir düzlemde bulunan iki grup arasında bir sınır çizilerek iki grubu ayırmak mümkündür. Bu sınırın çizileceği yer ise iki grubun da üyelerine en uzak olan yer olmalıdır. İşte sestek vektör makineleri (ing. Support Vector Machine) bu sınırın nasıl çizileceğini belirler. Bu işlemin yapılması için iki gruba da yakın ve birbirine paralel iki sınır çizgisi çizilir ve bu sınır çizgileri birbirine yaklaştırılarak ortak sınır çizgisi üretilir.

CIFAR-10 probleminde SVM kullanmak çeşitli doğruluk oranları elde edilmesini sağladı. Ancak bu doğrulun oranları, %33.5’u geçemedi. CNN ile kıyaslayınca aradaki bu büyük farkın sebebi, SVM’in görüde çok tercih edilen bir yöntem değilken, CNN’in bir şekilde her görü probleminde yer almasından kaynaklıdır.

|  |  |
| --- | --- |
| **Methods** | **Test Doğruluğu (%)** |
| No normalization on data | 30.6 |
| Normalization no. 1 | 30.6 |
| Best hyperparameters by tuning on validation set | 29.3 |

****

**Şekil 3.** MLP kullanırken kontrastın artırılması ve interpolasyon yapılması başarımı artırmıştır.

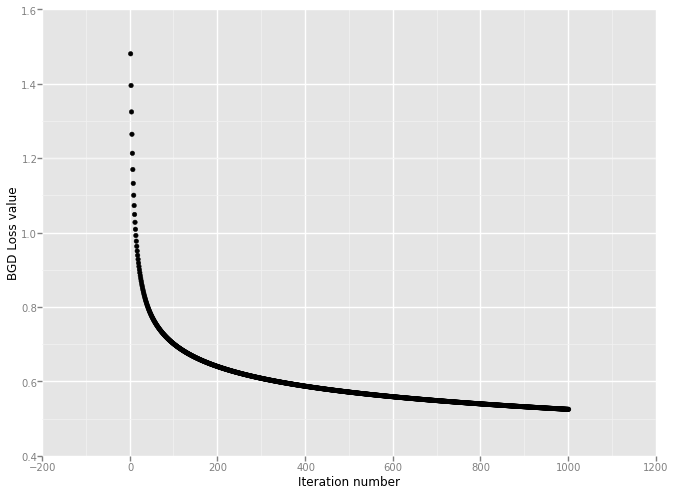
****

## Random Forest

Rastgele orman, sınıflandırma, regresyon ve diğer görevler için, eğitim aşamasında çok sayıda karar ağacı oluşturarak problemin tipine göre sınıf (sınıflandırma) veya sayı (regresyon) tahmini yapan bir toplu öğrenme yöntemidir. Rastgele karar ormanları, karar ağaçlarının eğitim setlerinde oluşabilecek aşırı öğrenme problemlerini gidermektedir. Bunu hem veri setinden hem de öznitelik setinden rastgele 10'larca 100'lerce farklı alt-setler seçerek ve bunları eğiterek başarıyor. Bu yöntemle 100'lerce karar ağacı oluşturuluyor ve her bir karar ağacı bireysel olarak tahminde bulunuyor. Sonuç olarak, problem regresyonsa karar ağaçlarının tahminlerinin ortalaması alınıyor, problem sınıflandırmaysa tahminler arasında en çok oy alan seçiliyor.

CIFAR-10 bir sınıflandırma problemi olduğu için, rastgele orman tercih edilebilir bir yöntem olarak görünüyordu. Diğer modellerdekine benzer ön işlemeler sonucu veri seti rastgele orman sınıflandırıcısında eğitilmiştir ve yaklaşık olarak %46.7’lik bir doğruluk oranı gözlemlenmiştir.

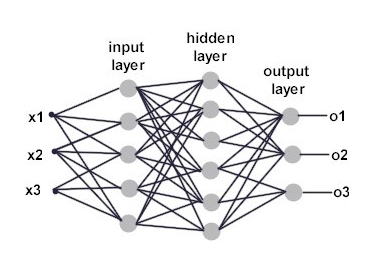
## Logistic Regression

****Lojistik regresyon, doğrusal sınıflandırma problemlerinde yaygın bir biçimde kullanılır. Regresyon denilmesine rağmen burada bir sınıflandırma söz konusudur. Çünkü, lojistik regresyon sınıflandırma işlemi yapmaya yarayan bir regresyon yöntemidir. Kategorik veya sayısal verilerin sınıflandırılmasında kullanılır. Bağımlı değişkenin yani sonucun sadece 2 farklı değer alabilmesi durumda çalışır. CIFAR-10 veri setinde bu sınıflandırıcıyı kullanırken SGD ve BGD kullanıldı. En iyi doğruluk oranları bu gradient descent metotlarıyla elde edildi. SGD ile loss çok daha hızlı azalırken BGD’de bu süreç daha yavaş ilerlemiştir. İkisinin de grafiğine aşağıda yer verilmiştir. Doğruluk oranları yakındır. Doğrulama seti (ing. Validation set) ve düzenleme (ing. Regularization) kullanıldığında, kullanılmadığına kıyasla %1’lik bir başarım artışı gözlenmiştir. Elde edilen en yüksek doğruluk oranı %82.9’dur ve CNN, yani en yüksek başarım elde edilen ve bir derin öğrenme modeli olan sınıflandırıcıya en yakın doğruluk oranına sahiptir. CIFAR-10 veri setinde makine öğrenmesi ile elde edilebilecek en yüksek doğruluk oranını lojistik regresyon vermektedir.

**Şekil 4** Lojistik regresyonda BGD ile iterasyona bağlı loss grafiği

**Şekil 5.** Lojistik regresyonda SGD ile iterasyona bağlı loss grafiği

## MLP

Bir sinir ağı, sadece farklı yollarla bağlanan ve farklı aktivasyon fonksiyonlarında çalışan perceptronların bir bileşimidir. Çok katmanlı algılayıcılar (ing. Multi-layer Perceptron), yapay sinir ağlarına olan ilgiyi hızlı bir şekilde artırmıştır. MLP, XOR problemini çözmek için yapılan çalışmalar sonucu ortaya çıkmıştır. MLP özellikle sınıflandırma ve genelleme yapma durumlarında etkin çalışır.

**Şekil 6**. MLP yapısının çalışma şekli resmedilmiştir.

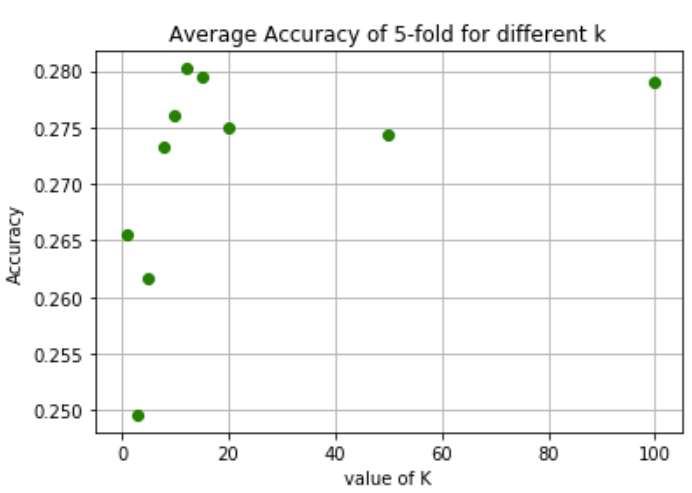
MLP kullanılarak eğitilen CIFAR-10 veri setinde, %50.3’lük bir doğruluk oranı elde edilmiştir. Aktivasyon fonksiyonu olarak RELU tercih edilmiştir ve validation doğruluğu %58 elde edildikten sonra bazı değerler düzeltildikten sonra fit edilmiştir ve nihai doğruluk oranına ulaşılmıştır. Bu düzenleme, görsellerin kontrastını artırmaktır ve interpolasyon yapmaktır. MLP için bu yöntemin daha faydalı olduğu görülmüştür ve optimal nöron büyüklüğü [7000,6000] olarak bulunmuştur.

## Naïve Bayes

Naïve Bayes sınıflandırma algoritması, adını matematikçi Thomas Bayes’den alan bir sınıflandırma algoritmasıdır. Naïve Bayes sınıflandırması olasılık ilkelerine göre tanımlanmış bir dizi hesaplama ile, sisteme sunulan verilerin sınıfını yani kategorisini tespit etmeyi amaçlar.

Naïve Bayes sınıflandırmasında sisteme belirli bir oranda öğretilmiş veri sunulur. Eğitim için sunulan verilerin mutlaka bir sınıfı bulunmalıdır. Öğretilmiş veriler üzerinde yapılan olasılık işlemleri ile, sisteme sunulan yeni test verileri, daha önce elde edilmiş olasılık değerlerine göre işletvilir ve verilen test verisinin hangi kategoride olduğu tespit edilmeye çalışılır. CIFAR-10 veri setinde de başarımı test edilmiştir. Naïve Bayes kullanarak elde edilen en yükse doğruluk oranı %29.7 civarında olmuştur. Bu oranla Naïve Bayes incelenen sınıflandırıcılar arasında en düşük başarımı elde eden modeller arasında olmuştur.

## KNN

k-en yakın komşu, denetimli öğrenmede sınıflandırma ve regresyon için kullanılan algoritmalardan biridir. En basit makine öğrenmesi algoritması olarak kabul edilir. Diğer denetimli öğrenme algoritmalarının aksine, eğitim aşamasına sahip değildir, bir başka deyişle eğitim yoktur (ing. No learning). Bu nedenle, geniş veri setini işlemek için gereken algoritma olarak ideal bir aday değildir. Yine de elimizdeki CIFAR-10 verisetini test etmek için kolay bir yöntem olduğu için araştırmada yer verilmiştir. %27.8’lik bir doğruluk oranı vermiştir test verisinde. Bu oran da bu araştırmada başvurulmuş modeller arasında en düşük doğruluk oranıdır. Fakat farklı deneylere göz atıldığında, kNN kullanarak CIFAR-10 üzerinde elde edilmiş en yüksek doğruluk oranının %38.6 olduğu görülmüştür. 

**Şekil 7.** K değişkeninin değerine bağlı olarak elde edilen doğruluk oranının grafiği

# **TEST SONUÇLARI VE YORULMARI**

Görsel sınıflandırma problemine birçok farklı sınıflandırıcı kullanarak farklı yaklaşımlarda bulunuldu ve farklı doğruluk oranları elde edilmiştir.

Bunlardan en düşük performansı gösteren en yakın komşu ile başlamak gerekirse, bunun sebebinin problemin doğası gereği özniteliklerince ayrıştırılabilir bir yapısı olmayışı söylenebilir. Burada pikseller öznitelik olarak düşünülürse en yakın komşunun iyi bir doğruluk oranına ulaşabilmesi için uzayın renkler cinsinden ayrılmış olması gerekirdi. Örneğin kurbağa sınıfının genellikle yeşil görsellerden oluşurken, araba sınıfının da genellikle beyaz görsellerden oluşması ve geriye kalan 8 sınıfın da kendine ait renk uzaylarına sahip olması gerekirdi. Bu durum tabii ki pratikte uygulanabilir bir yapı değil. Yine de doğruluk oranı %10’dan yüksek olduğu için rastgele tahminden çok daha iyi olduğu söylenebilir en yakın komşunun.

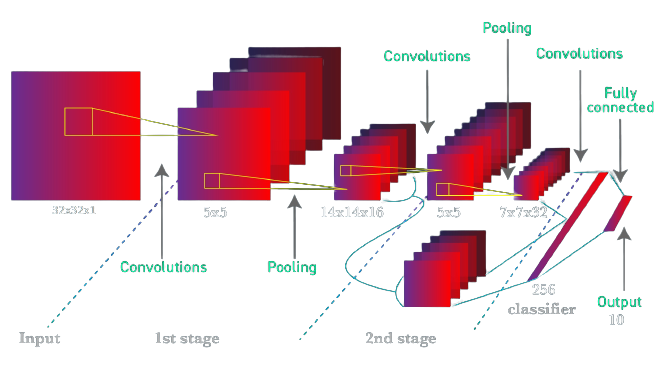
Lojistik regresyonun %82.9’luk test doğruluğu incelendiğinde, yapısı gereği lojistik regresyon [0,1] aralığında değerler döner ve bu değerler incelendiğinde en yüksek başarım [0,0.2] ve [0.8,1] aralığında gözlemlenmiştir. [0.3,0.7] aralığında ise genellikle yanlış sınıflandırmalar yer almaktadır. Test verisinde SGD veya BGD kullanmanın pek farkı olmadığı gözlemlenmiştir. Sadece BGD, SGD’ye göre loss değerinin yeterince küçülmesi için daha fazla iterasyona ihtiyaç duyar.

Rastgele orman, birçok karar ağacı oluşturur ancak sayılı öznitelik seçer aynı zamanda. Bunun nedeni hem aşırı öğrenmeden kaçınmaktır hem de hangi özniteliklerin karar merceği olduğunu ölçmektir. Veri setimiz görsellerden oluştuğu için özniteliklerin kullanılmayışı pek istenen bir yöntem değildir. Buna karşın %46.7’lik bir doğruluk oranı elde edilmiştir.

MLP, %50.3’lük doğruluk oranıyla tatmin edici bir sonuç vermiştir. Veri, ön işleme esnasında diğer sınıflandırıcılara kıyasla farklı süreçlerden geçmiştir. Multi-layer perceptron kullanırken, görselin kontrastını artırmanın doğruluk oranını artırabileceği gözlemlendi.

Destek vektör makinesi sınıflandırıcısında başarım %33.5 idi. Lojistik regresyonda olduğu gibi SVM’de de SGD ve BGD kullanılmıştır. Validasyon seti ile hiper parametreler oluşturulmuştur ve başarım bu seviyelere çıkabilmiştir. Öğrenme sınırlayıcı kullanılmıştır ve loss düşürülmüştür. Düzenleme (ing. Regularization) kullanılmıştır. Eğitim doğruluk oranı %15, validasyon doğruluk oranı %16 çıkmıştır.

CNN araştırma kapsamında en yüksek doğruluk oranını gözlemleyip kıyaslayabilmek için kullanılmıştır. En küçük katmanlı modelle bile %75 doğruluk oranlarına kolayca ulaşılır. Bazı optimizasyonlarla bu oran %85’e kadar çıkarıldı.



**Şekil 8.** 32x32’lik bir görselin CNN ile adım adım işlenme şeması ve çıktı olarak fully connected 10 sınıf ile sınıflandırması gösterilmiştir.

Naïve Bayes ile %29.7’lik bir başarım elde edilmiştir ve bu çalışmanın en düşük ikinci doğruluk oranına sahiptir Naïve Bayes. Sonuncu ise k en yakın komşudur ki onda da eğitim aşaması mevcut değildir. Bunun nedeni algoritmanın olasılık teoriminden gelmesi olabilir. Koşullu olasılığın yer aldığı Naïve Bayes formülü, görü prblemlerinde yüksek başarımlar elde edemez. K-en yakın komşuda da bahsedildiği gibi, öznitelikler görüde çok fazladır. CIFAR-10 veri setine bakıldığında 32x32 çok küçük görseller yer alır. Bu denli küçük görseller bile öznitelik uzayı için fazladır ve Naïve Bayes’in olasılıksal formülü görsellerde piksellerin uzaydaki dağılım şekliyle o kadar iyi uyuşmaz.

Lojistik regresyon, %82.9’luk doğruluk oranıyla, görü problemlerinde vazgeçilmez olan CNN modeline çok yaklaşıyor. Bunun nedeni sınıflandırıcının [0,1] aralığında sınıflandırma yapması ve veri ön işlemeden geçerken piksel değerlerinin de [0,1] aralığına indirgenmesi olabilir.

# **SONUÇLAR**

Çalışma sonucunda görü problemlerine nasıl bir yaklaşım sergilenmesi gerektiğine dair fikir edinildi. CNN’in bu tür problemlerde kaçınılmaz olduğu ve diğer sınıflandırıcılara kıyasla açık ara bir başarım farkı ortaya koyduğunu gördük. Bunun nedeni aşağıdaki görselde de görüldüğü üzere sistemin çıktı olarak istenen sayıda sınıftan birine girdiyi ilişkilendirmesidir.

Veri setiyle eğitimi gerçekleştirilen ve test verisiyle doğruluk oranları belirlenen sınıflandırıcılar, en iyiden en kötüye sırayla, CNN (%85), MLP (%50.3), Random Forest (%46.7), SVM (%33.5), Naïve Bayes (%29.7) ve kNN (%27.8) olmuştur.

# **REFERANSLAR**

1. Deep Learning with Python, François Chollet, Deep Learning for Computer Vision p. 119-177
2. Deep Learning with Python, François Chollet, Fundamentals of Machine Learning p. 93-116
3. tr.wikipedia.org
4. medium.com/analytics-vidhya/image-classification-techniques-83fd87011cac
5. medium.com/@ekrem.hatipoglu/machine-learning-classification-logistic-regression-part-8-b77d2a61aae1
6. machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/
7. towardsdatascience.com/cifar-10-image-classification-in-tensorflow-5b501f7dc77c
8. notebook.community/nishgaba-ai/Machine-Learning/CIFAR-10
9. github.com/vaibhavkumar11/Image-Classification-Using-CIFAR10
10. github.com/nishgaba-ai/Machine-Learning
11. github.com/osamakhaan/CIFAR-10-Image-Classification/blob/master/SVMs\_Part\_1.ipynb
12. houxianxu.github.io/implementation/SVM.html
13. isikhanelif.medium.com/multi-layer-perceptron-mlp-nedir-4758285a7f15
14. veribilimcisi.com/2017/07/20/k-en-yakin-komsu-k-nearest-neighborsknn

# **EK**

Aşağıdaki Github linki üzerinden çalışmanın yapıldığı Jupyter Notebook’a erişim sağlayabilirsiniz:

<https://github.com/safademirhan9/CIFAR-10_Review>