

Temsil Öğrenme için Bongard Problemleri

Bongard Problems for Representation Learning

Mehmet Fatih Amasyalı¹

¹Bilgisayar Mühendisliği Bölümü, Yıldız Teknik Üniversitesi, İstanbul, Türkiye
mfatih@ce.yildiz.edu.tr

Özetçe—Yapay sinir ağlarının son zamanlarda birçok alanda elde ettikleri başarının altında problemin çözümü için en anlamlı temsilleri / dönüşümleri bulmaları yatmaktadır. Bongard'ın makineler ve insanlar için önerdiği problemler, olası birçok temsil içinden doğrusunu bulma olarak tanımlanabilir. Bu çalışmada Bongard problemlerinin en alt seviye temsillerinin bir kısmının ağlarla elde edilebildiği gösterilmiş ve bu problemlerin yapay sinir ağları için bir sonraki hedef olabileceği önerilmiştir.

Anahtar Kelimeler — temsil öğrenme; yapay sinir ağları; derin öğrenme; konvolüsyonel sinir ağları; yapay zeka; yapay öğrenme; Bongard problemleri.

Abstract— The capability of finding the most appropriate representation is the key of ANN's recent successes on the several areas. Bongard introduced some pattern recognition problems. These problems can be defined as the finding the true representation of the shapes in the problems. In this study, some of the most basic representations of these problems can be identified with ANNs. We also suggest that Bongard problems can be the next challenge for the ANNs.

Keywords — representation learning; artificial neural networks; deep learning; convolutional neural networks; artificial intelligence; machine learning; Bongard problems.

I. GİRİŞ

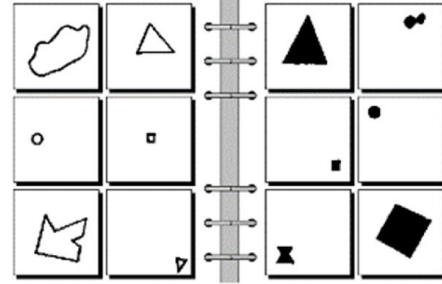
Yapay öğrenme literatürü incelendiğinde genel olarak ve doğal olarak algoritmaların (son zamanlarda derin öğrenme algoritmalarının) neler yapabildiğine odaklanıldığı görülmektedir. Ama büyük resmi gözden kaçırmamak için bu algoritmaların henüz neleri yapamadıklarına da bakmak gerekir. Bilinenle henüz bilinmeyenin sınırını görmek, olası çalışma alanlarını ortaya çıkarmanın en iyi yollarından biridir [1].

Yapay öğrenme algoritmaları şekil tanıma / işleme konusundaki en yüksek başarılarını, 1000 sınıflı, milyonlarca örnek içeren Imagenet yarışmasının (<http://image-net.org>) veri kümelerinde göstermiştir. Bu yarışmalardaki takımların büyük bir çoğunluğu çok katmanlı / derin sinir ağlarının bir çeşidi olan konvolüsyonel sinir ağlarını kullanmaktadır [2].

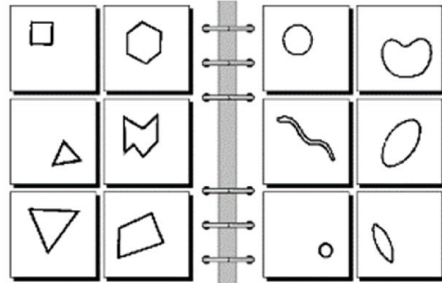
Derin öğrenmede tanıma, en alt basamakta basit, küçük ve ortak filtrelerle (şekil parçaları) başlamakta, hiyerarşik olarak ilerleyen üst basamaklarda ise daha büyük / kompleks şekil parçaları tanınmakta, en sonda ise sınıflar tanınmaktadır. Her probleme özgü temsilleri elle tasarlamak yerine, ağın probleme en uygun temsilleri kendisinin öğrenmesi sağlanmaktadır. Ağdaki temsiller birimler arası bağlantıların ağırlıklarında saklanmaktadır.

Bongard, 1970'te yayınladığı kitabında [3] insanlar ve makineler için 100 adet şekil tanıma problemi önermiştir. Her bir problemde 2 sınıfa ait 6'şar örnek verilmekte ve sınıfları birbirinden ayıran şeyin ne olduğu sorulmaktadır. Şekil 1'de örnek problemler verilmiştir.

BP#3. Designer: M. M. Bongard



BP#5. Designer: M. M. Bongard



Şekil 1. 3 ve 5 nolu Bongard problemleri

Şekil 1'deki 3 nolu problemin çözümü kolayca görülebileceği gibi sol sınıftaki tüm şekillerin içlerinin

dolu, sağ sınıftakilerin ise boş olmasıdır. Şekillerin büyüklük, şeklin türü, konumu gibi farklı temsillerinin bir önemi yoktur. 5 nolu problemde ise sınıflar şekillerin düz ya da eğri çizgilerden oluşmasıyla birbirinden ayrılmaktadır. Şekillerin diğer tür temsillerinin bir önemi yoktur.

Örneklerden görüldüğü gibi Bongard problemleri şekillerin nasıl temsil edildiğinin anlaşılmasını gerektirmektedir. Dolayısıyla bu problemler, bir şeklin çok sayıda olası temsilden doğru (sınıfları ayıran) olanının hangisi olduğunun bulunması olarak diğer bir ifade ile temsil öğrenme olarak tanımlanabilir [4].

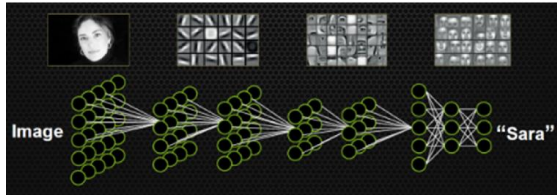
Sunulan bu çalışmada, temsil öğrenmenin bir makine tarafından yapılabildiğinin testi için, makine öğrenmesi algoritmalarının bir sonraki hedefinin Bongard problemleri olabileceği önerilmektedir.

Bildirinin devamında Bongard problemlerinin bu amaç için neden uygun olduğu açıklanacak ve yapılan deneyler sunulacaktır.

II. KONVOLÜSYONEL SİNİR AĞLARI

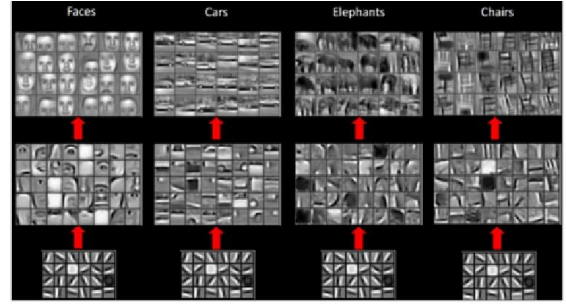
Şekil tanıma, klasik yapay sinir ağlarında alt parçalar yerine bütünü temsiline odaklanılmaktadır. Konvolüsyonel ağlarda ise önce şekli oluşturan alt şekiller bulunmakta, ardından bunlar ileri seviyelerde birleşmektedirler [5].

Şekil 2’de ağa bir yüz imgesi gelmekte, ilk aşamada basit şekillerin varlığını kontrol edilmekte, ilerleyen aşamalarda daha kompleks şekillerin (alt seviyedeki şekillerin birleşmesinden oluşan) varlığı kontrol edilmektedir.



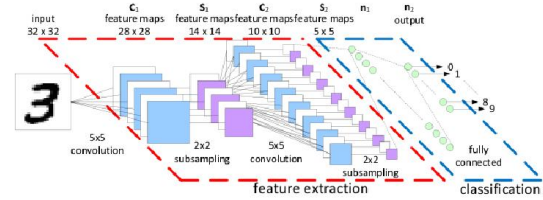
Şekil 2. Konvolüsyonel Sinir Ağlarıyla Kişi Tanıma [6]

Bu mimariye gerekçe olarak, görsel verinin insanda işlenmesinin de benzeri bir mekanizmaya sahip olması verilmektedir. Tüm şekiller küçük şekillerin birleşiminden oluşur. Buna örnek olarak Şekil 3’te nesne tanıma için oluşturulan bir ağın en alt seviyesindeki şekillerin aynı olduğu, yukarılara çıkıldıkça farklı nesne sınıfları için şekillerin değiştiği görülmektedir.



Şekil 3. Konvolüsyonel Sinir Ağlarıyla Nesne Tanıma [7]

Şekil 4’te bu ağların genel yapısı sayı tanıma örneği üzerinde gösterilmiştir.



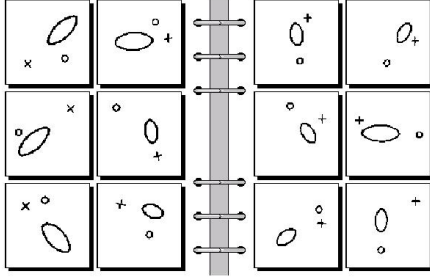
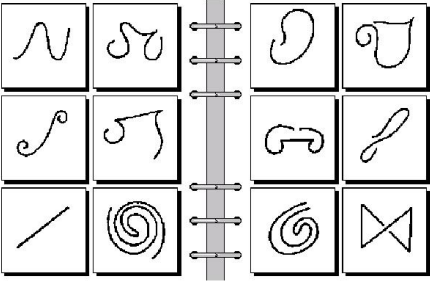
Şekil 4. Konvolüsyonel Sinir Ağlarıyla Sayı Tanıma [8]

Şekil 4’te görüldüğü üzere, ağda konvolüsyon, alt örnekleme (subsampling) ve klasik yapı (en son katman) bir aradadır. Konvolüsyon işleminde her bir birim kendinden önceki tüm birimlere bağlanmak yerine resimdeki bir bölgeye bağlanır, bu bölgede kendini arar. Kendini bulup bulamadığını bir sonraki katmana iletir. Alt örneklemede ise yine birim sadece bir bölgeye odaklanır ve buradan aldığı sinyalleri ortalama almak ya da maksimumunu almak gibi bir işlemde geçirerek sonraki katmana iletir. Bu işlemin amacı, bir şeklin çeşitli kaymalarına karşı yine de tanınabilmesini sağlamaktır.

Bu tür mimariler, son zamanlarda popülerlik kazanmış, üzerlerine akademik birçok çalışma yapılmış ve ticari uygulamalarda kullanılmaya başlanmıştır.

III. NEDEN BONGARD PROBLEMLERİ

Bongard problemleri mevcutta insan zekasını derecelendirmek için de kullanılabilen, henüz makineler tarafından çözülememektedirler [9, 10]. Şekil 5’teki problemler incelendiğinde, bunların makinelerce çözülebilmemesinin ne denli zor olduğu görülebilir.

BP#64. Designer: M. M. Bongard**BP#62. Designer: M. M. Bongard****Şekil 5.** 64 ve 62 nolu Bongard Problemleri

Günümüzde nesne tanıma yarışmalarında kullanılan veri kümelerinde çok sayıda sınıfa ait çok sayıda örnek yer almaktadır. Farklı sınıfların örnekleri alt seviyelerde benzer şekillerden oluşmakta, üst seviyelerde bu şekiller farklılaşmaktadır. Yüzlerce farklı nesne sınıfının tanınmasının bu yolla mümkün olduğu yarışma sonuçlarından görülmektedir [2].

Bongard problemlerindeki örnekler de ikili resimler halinde yapay öğrenme algoritmalarına kolaylıkla verilebilir. Orijinal problemler 2 sınıflı veri kümeleri olarak düşünülebilir. Eğitim ve test örnekleri olarak orijinal veriler ya da orijinal kümeyle aynı mantıkla oluşturulmuş yapay örnekler kullanılabilir.

Problemlerin çözümünü ifade eden sınıflar arası farklılıklar hiyerarşik olarak alt seviyedeki şekillerin birleşimi olarak elde edilebilir. Örneğin noktalar birleşerek çizgileri, çizgiler birleşerek şekilleri üretebilirler. Dolayısıyla, hiyerarşik mimarilerin örnekleri oluşturan şekillerin tanınması ve çeşitli özelliklerinin (yer, doğrultu, doku vb.) belirlenmesi için kullanımı mümkün görünmektedir. Ancak örnekleri oluşturan şekillerin çeşitli katmanlarda çeşitli temsilleri oluşturulduktan sonra bunların nasıl birleştirileceği, hangilerinin sınıfları birbirinden ayırdığı problemi ortaya durmaktadır. Burada mevcut mimarilerde yapıldığı gibi sadece alt seviye şekillerin birleşip yeni şekiller üretmenin yetmeyeceği açıktır. Örneğin 64 nolu problemde mevcut mimarilerle, noktalardan yola çıkılarak elipsler, çarpılar, daireler

tanınabilir, konumları bulunabilir. Ancak sınıfların, elipslerin gösterdiği yönlerdeki şeklin ne olduğuna (sol sınıfta çarpı, sağ sınıfta daire) göre ayrıldığına bulunması için nasıl bir mimari gerektiği henüz bilinmemektedir.

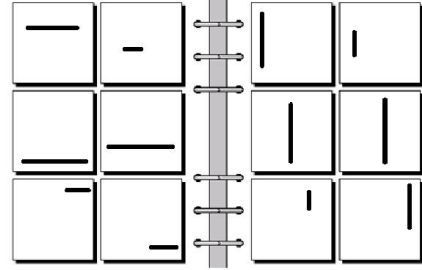
IV. MEVCUT MİMARİLER NELER YAPABİLİR / NELER YAPAMAZ?

İki katmana sahip klasik yapay sinir ağları gizli katmandaki birim sayısının sonsuza gidişiyse herhangi bir eğitim setini öğrenebilir [11]. Aslında, gizli katmandaki birim sayısının eğitim setindeki örnek sayısına eşit olması durumunda, her bir birim bir eğitim örneğine karşılık gelir ve tüm örnekler için eğitim setindeki çıkış elde edilebilir. Gizli katmanı giriş katmanına bağlayan ağırlıklarda şekiller depolanır. Klasik ağlarda bu birimler tüm resme bağlı iken, konvolüsyonel ağlarda resmin bir kısmına bağlıdır.

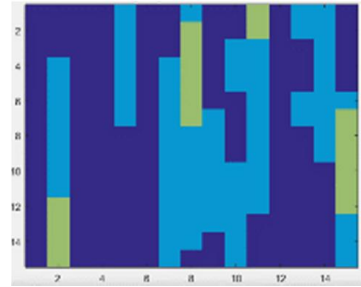
Bu bölümün devamında Bongard problemlerinin en alt seviyesindeki bazı temsilleri öğrenmede klasik ağlar ve konvolüsyonel ağlar karşılaştırılacaktır.

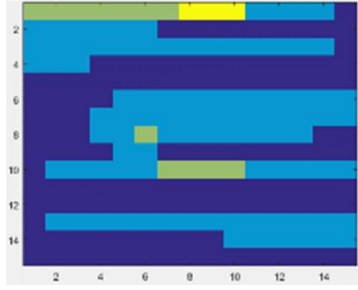
A. Yatay Dikey Çizgi Ayırma

Klasik ağların dikey bir çizgiyi, yatay bir çizgiden ayırıp ayıramayacağını görmek için bir deney tasarlanmıştır. Bu problem bir Bongard problemi olarak Şekil 6'daki gibi ifade edilebilir.

**Şekil 6.** Yatay ve dikey çizgileri ayırmanın bir Bongard problemi olarak temsili

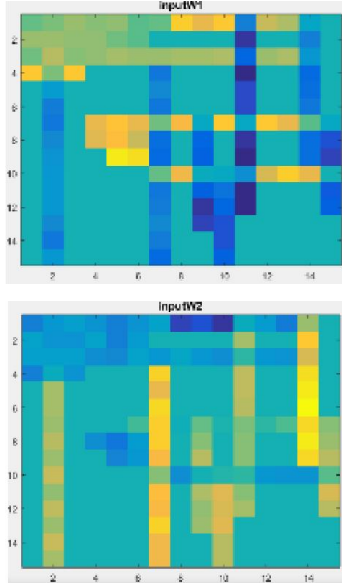
15*15'lik ikili resimler üzerine rastgele uzunluklarda dikey ya da yatay 32 çizgi çekilerek 2 sınıfa ait 32 örnek üretilmiştir. Şekil 7'de üretilen çizgilerin 15*15'lik resim üzerinde işgal ettikleri hücrelerin frekansları görülmektedir.





Şekil 7. Yatay ve dikey çizgiler veri kümesi

Üretilen veri kümesi 16 örneklik 2 parçaya ayrılmış, bu parçalar eğitim ve test kümeleri olarak kullanılmıştır. 2 katmanlı, gizli katmanda 2 birim bulunan bir ağ, Levenberg-Marquardt ile eğitilmiştir. Şekil 8’de bu eğitim sonucu giriş katmanı ile gizli katmandaki 2 birim arasındaki ağırlıklar gösterilmektedir.



Şekil 8. Yatay ve dikey çizgiler veri kümesi için üretilen ağırlıklar

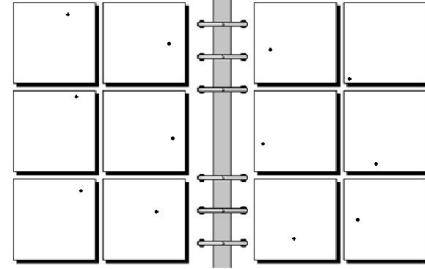
Ağırlıklar incelendiğinde birbirinin tersi oldukları görülmektedir. Ayrıca eğitim örnekleri de ağırlıklar içinde kolaylıkla seçilebilmektedir. Şekil tanıma için kullanılan yapay sinir ağları için kullanılan kara kutu benzetmesinin yanlış olduğu burada görülmektedir. Gizli katmanla giriş katmanı arasındaki ağırlıklar eğitim örneklerini saklamaktadır. Eğitim kümesindekilere benzer bir test örneği geldiğinde şekli tanıdığı sinyali bir sonraki katmana iletmektedirler.

Bu veri kümesi için eğitim kümesi ezberlenmiş (hata=0), test kümesi ise %50 başarı ile tahmin edilebilmiştir. Diğer bir deyişle öğrenme mümkün olmamıştır. Ağın gizli katmanındaki birim sayısının artışı, örnek sayısının artışı bu durumu yaptığımız denemelerde değiştirmemiştir.

Klasik sinir ağlarının bu durumuna rağmen, bir konvolüsyonel ağ bu sınıfları ayırmayı kolayca öğrenebilir. Bunun için resim üzerinde 3*3’lük dikey ve yatay kenar belirleyen filtreler gezdirmesi yeterli olacaktır.

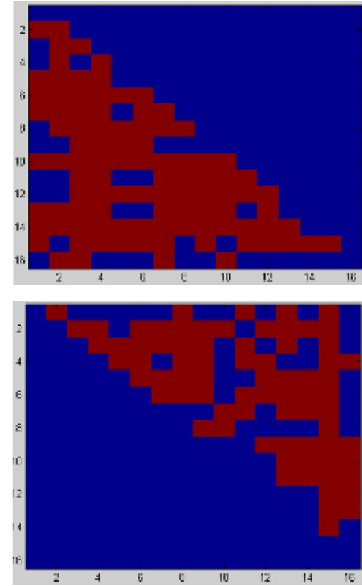
B. Noktaları Diyagonalin Altında ve Üstünde Oluşuna Göre Ayırma

Bu denemede 16*16’lık bir resim üzerindeki bir noktanın resmin sol alt kısmında mı, sağ üst kısmında mı olduğu bulunmaya çalışılmıştır. Bu problem bir Bongard problemi olarak Şekil 9’daki gibi ifade edilebilir.



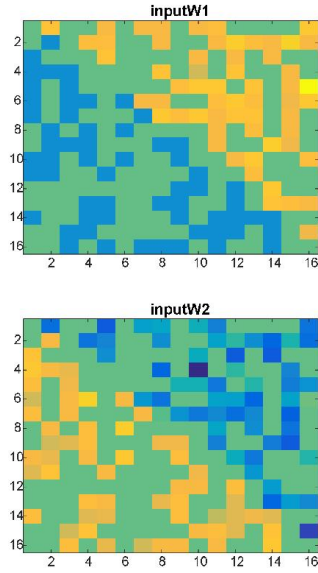
Şekil 6. Noktaları diyagonalin altında ya da üstünde olmalarına göre ayırmanın bir Bongard problemi olarak temsili

Bu problem, noktalar resim üzerinde bir nokta değil de, iki adet koordinatı ile verilirse oldukça kolay (lineer) bir problem olacaktır. Şekil 10’da üretilen 2 sınıfa ait 320 verinin birleşik halleri verilmiştir.



Şekil 10. Diyagonalin altı üstü veri kümesi

Bu problem içinde yine 2 katmanlı ağ kullanılmış, 160’şar örnek eğitim ve test kümesi olarak kullanılmıştır. Şekil 11’de elde edilen ağırlık değerleri görülmektedir.

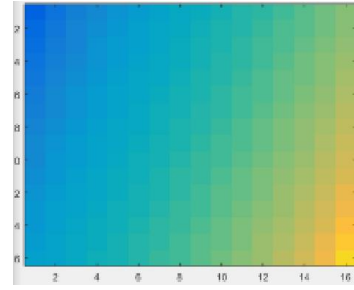
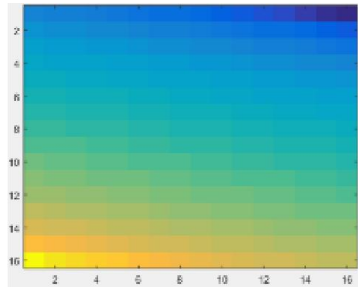


Şekil 11. Diyagonalın altı üstü veri kümesi için üretilen ağırlıklar

Ağırlıklar incelendiğinde yine eğitim örnekleri fark edilmektedir. Sistem yine eğitim kümesini ezberlemiş, test kümesinde rastgele başarının ötesine geçememiştir. Bunun sebebi ağırlıklarda da görüleceği üzere boşluklardır. Eğitim kümesinde hiç bulunmayan hücreler için doğru tahminler üretememektedir. Bununla birlikte örnekler Gaussian filtre uygulandığında, noktalar birer bölge haline gelmekte, eğitim ve test örnekleri birbirine daha çok benzemekte ve dolayısıyla testte de başarı (%98) sağlanmaktadır. Gaussian filtrenin etkisi, konvolüsyonel ağlarının bu filtreyi öğrenip, bu problemi de çözebileceklerini göstermektedir. Bununla birlikte konvolüsyonel ağlarının alt örnekleme katmanı da bu problemin çözümü için yeterli olacaktır.

C. Nokta Koordinatı Bulma

Bu denemede 16×16 'lık bir resimde rastgele üretilen noktaların koordinatlarının bulunması amaçlanmıştır. Problem, 2 çıkışlı bir regresyon problemidir. Üretilen 3200 örneğin 1600'ü ile eğitilmiş klasik ağın ilk katmanının ağırlıkları Şekil 12'de verilmiştir. Bu ağırlıkların eğitim ve test örnekleri üzerindeki hatası 0'a yakındır.



Şekil 12. Nokta koordinatı bulma için üretilen ağırlıklar

Bu denemede olası farklı örnek sayısı $16 \times 16 = 256$ 'dır. 3200 örnekle elde edilen yukarıdaki başarıya eğitim ve test örneklerindeki tekrarlar sebep olmuştur. Diğer bir ifade ile test örneklerinin birer kopyası eğitim kümesinde de yer almaktadır. Tekrarlar kaldırıldığında (256 örneğin yarısı ile sistem eğitildiğinde) başarı da ortadan kalkmaktadır. Bununla beraber bu problemde de Gaussian filtre uygulandığında tekrarsız veriler üzerindeki test hatasının düşük olduğu görülmüştür.

D. Çizgi Uzunluğu Bulma

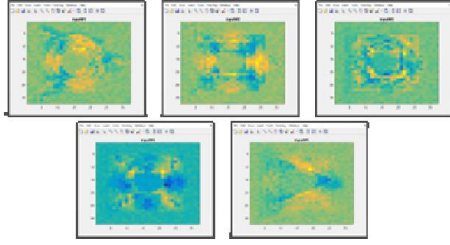
Bu denemede 16×16 'lık resim üzerinde rastgele uzunlukta dikey yatay bir çizgi çizilmiş, sistemin bu çizginin uzunluğunu bulması istenmiştir. Problem 1 çıkışlı bir regresyon problemidir. İlk bakışta zor gibi görünen problemi klasik bir ağ tüm ağırlıkları eşit yaparak, çıkışta lineer bir aktivasyon fonksiyonu ile çözebilmıştır. Aktif olan hücre sayısı çizginin uzunluğudur.

E. İki Nokta Arasındaki Manhattan Uzaklığını Bulma

Bu denemede 16×16 'lık bir resim üzerine rastgele yerleştirilen 2 noktanın arasındaki Manhattan uzaklığını sistemin bulması istenmiştir. 3200 örnek oluşturulmuş, yarı yarıya eğitim ve test kümesi olarak kullanılmıştır. Bu problemde noktaların birbirinden farklı olduğunu varsayarsak yaklaşık 32 bin olası örnek bulunmaktadır. 2 gizli birime sahip klasik bir ağ eğitim verisini bile öğrenememiştir. 10 gizli birimle eğitim öğrenilmiş ancak test başarısı düşük kalmıştır. Gaussian filtre uygulandığında ise test başarısı artmıştır. Ancak yine 10 gizli birim gerekli olmuştur. 320 örnek ile oluşturulan veri kümesinde ise test başarısı Gaussian filtre ile bile düşüktür.

F. Şekil Tanıma

Bu denemede öncelikle Bengio'nun [12] kullandığı 32×32 'lık bir resim üzerine yerleştirilen şeklin 3 sınıftan (kare, üçgen ve daire) hangisine ait olduğunu belirlenmesi üzerine çalışılmıştır. Veri kümesinde 5000 örnek bulunmakta ve her 3 sınıfın frekansları yaklaşık eşittir. Denemelerde 1024 girişi, 5 gizli birimli ve 3 çıkışlı ağlar kullanılmıştır. 250 örnek eğitim, 4750 örnek test için kullanılmıştır. Ağ, ölçeklenmiş eşlenik gradyan (scaled conjugate gradient) ile 1000 devir (epoch) eğitildiğinde eğitimde %100, testte %94.1 başarı elde edilmiştir. Ağ ağırlıkları Şekil 11'de gösterilmiştir.



Şekil 11. Şekil tanıma için Bengio'nun veri kümesi [XX] için üretilen ağırlıklar

Şekil 11 incelendiğinde ağırlıklarda prototiplerin belirgin olduğu görülmektedir. Örneğin 1. ve 3. daireyi, 2 ve 4. kareyi, 5. üçgeni ifade etmektedir. Test sonucu oldukça başarılıdır. Bununla birlikte veri kümesi incelendiğinde sınıf içi varyasyonların çok az olduğu görülmüştür. Tüm şekiller matrisin ortasına yerleştirilmiş, şekillerde açılı farkı bulunmamaktadır. Sınıf içi varyasyon sadece arka plan ön plan renk farklılıklarını ve az miktarda şekil boyut farklılıklarını içermektedir.

Bunun üzerine sınıf içi farklılıkları daha fazla olan bir veri kümesi oluşturulmuştur. Bu veri kümesinde yine 3 sınıf (daire, üçgen ve dikdörtgen) bulunmakta ve şekiller 50*50'lik bir matris üzerine minimum alanları 250 olacak şekilde çizdirilmiştir. Daire sınıfı için yer, büyüklük varyasyonları, üçgen sınıfı için yer, büyüklük ve açılı varyasyonları, dikdörtgen için yer ve büyüklük varyasyonları üretilmiştir. Şekillerin ortaları Bengio'nun veri kümesinde olduğu gibi doldurulmuştur. Şekillere dahil olan hücreler 1, arka plan hücreleri ise -1 olarak ifade edilmiştir. Eğitim ve test kümelerinin boyutları eşit olarak üretilmiştir. Bu veri kümesinde test kümesi üzerinde %89 başarı elde etmek için eğitimde kullanılan örnek sayısını 8000'e, gizli birim sayısını 100'e çıkarmak gerekmiştir.

Bu veri kümesi için ayrıca 2 otomatik kodlayıcı (autoencoder) + 1 üstel yumuşatma (softmax) katman kullanılarak bir derin bir yapay sinir ağı oluşturulmuştur. 8000 örneklilik eğitim kümesinde %94.5 test başarıları elde etmek için otomatik kodlayıcılardaki gizli birim sayılarını 100 ve 25'e çıkarmak gerekmiştir. Bu ağı parametre sayısı içerdiği fazladan katmanlar sebebiyle klasik ağıdan fazladır.

Sonuç olarak, klasik ağı yerine daha fazla parametreye sahip bir derin ağı ile aynı boyuttaki bir veri kümesi üzerinde daha başarılı test sonuçları elde edilebilmiştir. Bu sonuçların genelleştirilebilmesi için daha çok deneme yapılması gerektiği de açıktır. Bununla birlikte literatürde buna benzer sonuçlar elde edildiği görülmektedir [13].

V. TARTIŞMA VE AÇIK SORULAR

Bir önceki bölümde Bongard problemlerinin en alt seviye temsillerinden bazılarının çeşitli yapay sinir ağı mimarileriyle bulunabileceği gösterilmiştir. Klasik sinir ağları yerine konvolüsyonel ağlar daha fazla umut vaatmektedir. Bu alt seviye temsiller bir şekilde birleşerek

üst seviye temsillere dönüşebilir. Bu sebeple hiyerarşik mimari bu problemler için de uygun görünmektedir. Ayrıca derin ağlarında klasik ağlardan daha başarılı olduğu şekil tanıma problemi üzerinde görülmüştür.

Ancak bir makinenin Bongard problemlerini çözebildiğini iddia etmek için daha önce hiç görmediği bir temsili içeren sorulara da cevap verebilmesi gerekir. Diğer bir ifade ile mevcut 100 Bongard probleminin hepsinin cevabı incelenip, mevcut alt ve üst temsillerin hepsini bulabilen bir sistem muhtemelen tasarlanabilir. Örneğin 62 nolu problemin cevabını bildikten sonra sadece uç noktaları bulan bir filtre sisteme eklenip uçlar arasındaki mesafenin bulunması sağlanabilir. Ancak buradaki soru bu sistemin 100 problem içinde yer almayan alt ya da üst seviye bir temsil içeren bir problemi çözüp çözemeyeceğidir. En alt seviye temsillerin çok fazla çeşidi olmayabilir ancak üst seviye temsillerin sayısı tüketilebilir olmayabilir. Bir diğer sorun ise üst seviye temsillerin alt seviye temsillerden nasıl elde edileceğidir.

Henüz yapılamayanlarla yapılabilenler arasında bir noktada yer alan ve gerçek bir temsil öğrenme problemi olan Bongard problemleri yapay öğrenme için bir sonraki hedef olabilir. Bu da mevcut problemlerin nicelikselsel (sınıf / örnek / boyut sayısı) versiyonlarını çözmek yerine nitelikselsel bir değişim anlamına gelmektedir.

KAYNAKÇA

- [1] Firestein, Stuart. Ignorance: How it drives science. OUP USA, 2012.
- [2] <http://image-net.org/challenges/LSVRC/2014/index>
- [3] Bongard, M. M., Pattern Recognition. Rochelle Park, N.J.: Hayden Book Co., Spartan Books, 1970.
- [4] Hofstadter, D. R., Gödel, Escher, Bach: an Eternal Golden Braid. New York: Basic Books, 1979.
- [5] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time-series. In M. A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*. MIT Press, 1995.
- [6] <http://devblogs.nvidia.com/parallelforall/accelerate-machine-learning-cudnn-deep-neural-network-library>
- [7] <http://stats.stackexchange.com/questions/146413/why-convolutional-neural-networks-belong-to-deep-learning>
- [8] <http://www.es.ele.tue.nl/~dshe/Education/Dmm2012>
- [9] Foundalis, H., Phaeaco: A Cognitive Architecture Inspired by Bongard's Problems. Doctoral dissertation, Indiana University, Center for Research on Concepts and Cognition (CRCC), Bloomington, Indiana, 2006.
- [10] <http://www.foundalis.com/res/solvprog.htm>
- [11] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition", IEEE Trans. Electrom. Comput., Vol. 14, no.3, pp.326-334, 1965.
- [12] Bengio, Y., Louradour, J., Collobert, R., ve Weston, J. Curriculum learning. In Proceedings of the 26th annual international conference on machine learning, sayfa. 41-48, ACM, 2009.
- [13] Bengio, Y. Learning deep architectures for AI. Foundations and trends® in Machine Learning, 2(1), sayfa 1-127, 2009.