Journal of Information Systems and Management Research
Bilişim Sistemleri ve Yönetim Araştırmaları Dergisi

http://dergipark.gov.tr/jismar

Literatür Makalesi / Review Article

Makine Öğrenmesinde Yeni Bir Bakış Açısı: Otomatik Makine Öğrenmesi (AutoML)

Şebnem ÖZDEMİR[*,a,b], Suat ÖRSLÜ [c]

[a,*] *İstinye Üniversitesi, Yönetim Bilişim Sistemleri Bölümü, ISTANBUL, 34010, TÜRKİYE*
[b] *Massachusetts Institute of Technology, Computer Science&Artificial Intelligence Lab (CSAIL), BOSTON, 02139, USA*
[b] *EVP-Business Development / R&D EreTeam, ISTANBUL, 34893, TÜRKİYE*

MAKALE BİLGİSİ

Alınma: 31.10.2019
Kabul: 25.12.2019

Anahtar Kelimeler:
AUTOML, Makine Öğrenmesi, Veri

Sorumlu Yazar
e-posta:
sebnem.ozdemir@istinye.edu.tr

ÖZET

Veriden değer çıkartma sürecinde, kaliteli bir makine öğrenmesi yaşam döngüsünün sağlanması, sağlıklı verinin eldesi kadar, doğru araç ve doğru insan işbirliğine de bağlıdır. Teknolojik gelişmeler pek çok yeni ve başarılı aracı bu döngü için kullanılabilir hale getirmişse de yetkin insan sayısının azlığı önemli bir darboğaz yaratmaktadır. Otomatik Makine Öğrenmesi (AutoML), bu darboğazın aşılmasında, insan deneyimine bağlı sürecin daha bağımsız ve demokratik hale getirilmesi için kullanılmaktadır. Bu çalışmada, AutoML kavramına, geliştirilen araçlardaki temel yaklaşımlara yer verilmiştir. Ayrıca açık kaynaklı, startup destekli ve teknoloji devleri tarafından geliştirilen bazı araçların kapsamları hakkında da bilgi verilmektedir. Çalışmada AutoML'in insan işbirliği ile elde edebileceği başarı, bir veri seti ve üç takım üzerinden yapılan deneme süreci kapsamında sunulmaktadır. Elde edilen sonuçlar, makine öğrenmesi yarışmaları düzenleyen Kaggle'ın Mayıs 2019 tarihinde düzenlediği autoML – insan yarışmasıyla da uyumludur.

New Perspective on Machine Learning Process: AutoML

ARTICLE INFO

Received: 31.10.2019
Accepted: 25.12.2019

Keywords:
AUTOML, Machine Learning, Data

Corresponding Authors
e-mail:
sebnem.ozdemir@istinye.edu.tr

ABSTRACT

In the process of extracting value from the data and having quality in machine learning life cycle depend on the right tool and the right human cooperation as well as obtaining convenient data. Although technological advances have made many new and successful tools available for this cycle, the scarcity of competent people create a major bottleneck. Automated Machine Learning (AutoML) was suggested not only overcoming that bottleneck but also creating the process with the less human effect, more independent and democratic. In this study, the concept of AutoML and the fundamental approaches and the developed tools are given. The accuracy rates, obtained by AutoML with human collaboration, is also presented within the scope of a data set and trials by three teams. The results obtained are consistent with Kaggle's AutoML - human competition in May 2019.

1. GİRİŞ *(INTRODUCTION)*

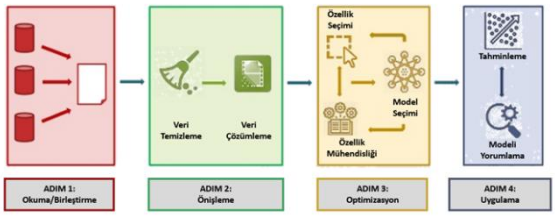
90'lı yıllardan itibaren verinin önemi giderek artmış, devletler, şirketler politikalarını veri odaklı hale getirmeye başlamışlardır. Yükselen bu değere karşın, o dönemlerde veri işlemede araç, amaç, kapsam ve doğru insanlarla çalışma gibi zorluklar bulunmakta idi. İçinde bulunduğumuz çağda elde edilen teknik gelişmelerle araç sorunu büyük ölçüde ortadan kalkmışsa da, amaç ve kapsam kısımları veriyi analiz edebilecek ekibin deneyimi ve kalitesiyle yakından ilgili olmasından dolayı hala sıkıntılıdır. Bu sıkıntının kaynağında özellikle makine öğrenmesi temelli çalışmalardaki etik, açıklanabilirlik, hesap verebilirlik bulunmakla birlikte, istenilen düzeyde bilgi ve beceriye sahip birey sayısının azlığı da yer almaktadır. Bu azlık, alanda eğitim gören, mezun olmuş, çalışan insan sayısındaki artışa karşın, sektördeki beklentiyi karşılama açısından yıllar geçtikçe daha kritik bir darboğaz yaratmaktadır [1-4]. Bu darboğaz; araştırmacıları ve büyük şirketleri süreci daha otomatize ederek, duyulan ihtiyacı daha aza indiren çözümler bulmaya yönlendirmiştir.

Özellikle son dönemdeki insanın bulunduğu her sistemde olduğu gibi, makine öğrenmesi sürecinde de insan hatasının oluşturduğu kritik sonuçlar, deneyime bağlı olarak performansı/kurgusu daha iyi ya da daha kötü modeller oluşturulması, önyargılara bağlı olarak modelde bias (önyargı, sapma, yanılgı) meydana gelmesi de bu çözüm bulma sürecini hızlandırmıştır. Bu sürece yönelik önemli bir çözüm olarak Otomatik Makine Öğrenmesi (AutoML) önerilmektedir [5].

AutoML matematiksel tarafıyla büyük ve çok yönlü bir optimizasyon problemi gibi düşünülebilmektedir. Bu problem; insan müdahili olmadan belli bir hesaplama bütçesine göre bir veri seti için tahminler üretebilecek şekildeki bir çözümler uzayıyla ifade edilmektedir. $i = 1, \dots, n+m$ olmak üzere $x_i \in R^d$'ler özellik vektörü ve $y_i \in Y$'ler ilgili hedef değeri olsun. $D_{egitim} = \{(x_1, y_1), \dots, (x_n, y_n)\}$ bir eğitim veri kümesi, $D_{test} = \{(x_{n+1}, y_{n+1}), \dots, (x_{n+m}, y_{n+m})\}$ test kümesi ve $x_{n+1}, \dots, x_{n+m}$ bu test kümesinin özellik vektörleri olmak üzere, belli bir b bütçesi ve kayıp ölçütü a karşısında benzer veri dağılımı çizmektedirler. Bu durumda $L(.,.)$ bir autoML problemi için test kümesi üzerinde $\hat{y}_{n+1}, \dots, \hat{y}_{n+m}$ tahminler oluşturulabilecektir. Bu tahminler karşısında oluşan kayıp da Denklem 1'de tanımlıdır [6].

$$1/m \sum_{j=1}^{m} L(\hat{y}_{n+j}, \dots, \hat{y}_{n+j}) \quad (1)$$

Diğer bir deyişle AutoML; bir veri seti üzerinde optimum performansı yakalayabilmek için makine öğrenmesi algoritmalarının, ön işleme adımından modelin işleme alınmasına kadar ki tüm adımlarının otomatik bir biçimde yürütülmesidir. [7] tarafından tanımlanan en genel biçimdeki autoML süreci ve adımlarının Türkçe'ye çevrilmiş hali Şekil1'de verilmektedir [7].



Şekil 1 AutoML süreci ve adımları
(The process and steps of AutoML)

AutoML olarak ifade edilen otomatikleştirilmiş bu sürecin başarısı hiperparametrelere bağlıdır. Her makine öğrenmesi çalışmasında hiperparametrelere (HP) bulunmakta olup, analiz sürecindeki temel görev bu HP'ler aracılığıyla makinenin performansını optimize etmektir.

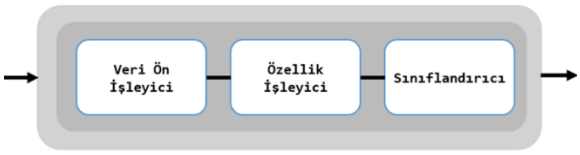
Hiperparametre optimizasyonu (HPO) olarak adlandırılan bu temel görev; aynı zamanda makine öğrenme sürecinin önemli bir problemidir [8-9]. Çünkü HPO, farklı veri kümelerinde en iyi şekilde çalışabilecek hiperparametre konfigürasyonlarını keşfedebilme [10], belli alanlarda kullanılabilir ardışık düzenlerin (pipeline) kurgusunu tasarlayabilme [11] ve ortak makine öğrenmesi kütüphanelerinde/paketlerinde varsayılan olarak sunulanların geliştirilmesine [12-15] katkı sağlayabilmek amacıyla kullanılmaktadır.

AutoML, HPO problemini, bireyin deneyimi doğrultusunda yapacağı pek çok deneme yanılma sürecinin maliyetini de ortadan kaldırarak makine öğrenmesi sürecini düzgün bir iş akışı halinde kullanabilmesini amaçlamaktadır [16-19]. AutoML kapsamında otomatikleştirilmiş adımları şu şekilde sıralamışlardır:

- Otomatikleştirilmiş veri hazırlama
  - Sütun bazlı nitelik türlerinin tespiti (sayısal, kategorik, ikili vb)
  - Sütun bazlı nitelik rolünün tespiti (hedef nitelik mi?)
  - Görev tespiti (ikili sınıflandırma, regresyon, kümeleme vb)
- Otomatik özellik mühendisliği
  - Özelliklerin seçimi ve çıkarımı
  - Meta öğrenme ve transfer öğrenme
- Otomatik model seçimi
- Algoritmanın öğrenmesinde Hiperparametre optimizasyonu
- Otomatik ardışık düzen seçimi (zaman hafıza ve kısıtlardaki karmaşaya göre)

- Değerlendirme metriklerinin ve validasyon prosedürlerinin otomatik seçimi
- Otomatik problem kontrolü
    - Sızıntı tespiti (Leakage detection)
    - Yanlış konfigürasyon tespiti
- Elde edilen sonuçların otomatik olarak analizi
- Makine öğrenmesi süreci için kullanıcı arayüzleri ve görselleştirme

Henüz taze olan ve geliştirilmeye devam eden autoML; literatürdeki örneklerine bakıldığında etkileyici sonuçlar ürettiği görülmektedir [18-19]. AutoML çalışmalarında, yöntemlerinde sistemin nasıl çalıştığının açıklanabilmesi oldukça önemlidir. Çünkü otomatikleştirilmiş bu süreç, hem kurgulanan modelin başarısının sorgulanabilirliğini hem de yapay zekadaki açıklanabilirlik kavramını karşılayabilir olmalıdır. Bu amaçla AutoML çalışmalarında sürecin nasıl ilerlediğini gösteren ardışık düzenler (pipelines) muhakkak tanımlanmalıdır. Şekil 2'de klasik bir AutoML sürecinin ardışık düzeni verilmektedir.



Şekil 2. Klasik AutoML lineer ardışık düzeni
(Linear pipeline of the classic AutoML)

Klasik bir AutoML ardışık düzeninde CRISP-DM, SEMMA gibi veri analizi süreç yönetiminde kritik olarak öne sürülen veri ön işleme adımı bulunmaktadır. Bu adımda eksik veriler, uç değerler, ölçeklendirme gibi temel işlemler gerçekleştirilmekte, veri bir nevi gürültüden arındırılmaktadır. İkinci aşamada ise özelliklere/niteliklere dair ön işleme süreci gerçekleştirilmektedir. Bilindiği üzere makine öğrenmesi modelleri gerçek dünyayı en iyi temsil edecek şekilde kurgulanmaktadır. Bu kurguda modeli oluşturan niteliklerin sayısının artması sanıldığı gibi daha hassas değil daha karmaşık bir model kurgusu yaratmaktadır. Boyutsallık laneti olarak adlandırılan bu durum, özellik çıkarımını/eliminasyonunu zorunlu kılmaktadır. AutoML'de de bu noktaya yer verilmiş olup modelin hangi nitelikler üzerinden inşa edilmesi gerektiği özellik önişleme adımında ortaya çıkarılmaktadır.

Son adım olan sınıflandırıcı adımı da ilgili özellikler doğrultusunda veriden öğrenen modelin oluşturulması eylemlerini barındırmaktadır. Çok temel bir yapıya sahip olan bu ardışık düzen AutoML'de tanımlanan yöntemin kapsamına göre özelleştirilmektedir.

1.1. AutoML Yaklaşımları *(AutoML Approaches)*

AutoMLde ardışık düzenin doğru kurgulanması, HP'lerin doğru bir şekilde yerleştirilmesi makinenin başarısında önemli bir etkendir [20]. Literatürde sıklıkla kullanılan dört önemli yaklaşım bulunmaktadır:

- Bayezyen Optimizasyon (Bayesian Optimization-BO): BO yöntemi girdi verisine aday modeller uygulanmasına dayalıdır. Yöntem, olasılıklı vekil modeli (probablistic surrogate model) ve yakalama fonksiyonu (acquisition) gibi iki temel bileşeni olan iteratif bir algoritma gibi hareket etmektedir. Bu bileşenler sayesinde kurgulanan döngüde, vekil modeli, hedef fonksiyonun ürettiği gözlemlere uyumlaştırılmış bir model olarak üretilmektedir [21-22].
- Meta Öğrenme (Meta Learning): BO yönteminden farklı olarak bu yöntemde verinin doğası üzerinden meta-özellikler kümesi oluşturulmaktadır. Bu meta özellikler, modelin gerçek anlamda eğitilmeksizin benzer bir veri seti üzerindeki geçmiş deneyimlerine dayalı performansından anlam çıkarabilmek üzere kullanılmaktadırlar [23].
- Evrimsel Algoritmalar (Evolutionary Algorithms-EA): HPlerin dağılımlarından ve bu dağılımların güncellenmesinden öğrenmeye dayalı bir yöntemdir.
- Pekiştirmeli Öğrenme Yaklaşımları (Reinforcement Learning Approaches-RL): Hiperparametre optimizasyon probleminin, hiperparametre uzayında hareket ederek ve çözümlerin RL teknikleri kullanılarak üretilmesine dayalı bir öğrenme politikası olarak formulize edilmesini içermektedir [24].

Bu yaklaşımları baz alarak kendi ardışık düzenlerini oluşturan çok çeşitli AutoML araçları bulunmaktadır. Farklı güç ve altyapı kurgusuna sahip bu araçların tamamı klasik makine öğrenmesi algoritmalarında başarılı sonuçlar üretmektedirler.

1.2. AutoML Araçları *(AutoML Tools)*

AutoML araştırmaları, özellikle HPO odaklı olanlar, 1990'lı yıllarda başlamıştır. Matematiksel altyapısıyla birlikte bir araç olarak geliştirilmesi 2010 yılından sonra mümkün olabilmiştir. Bu alandaki araçlar, açık kaynaklı olanlar, startuplar tarafından geliştirilenler ve teknoloji devleri tarafından sunulanlar olmak üzere üç ana grupta toplanmaktadırlar.

1.2.1. MLBox *(MLBox)*

MLBox, AutoML düşüncesiyle kurgulanmış bir Python kütüphanesidir. Bu kütüphanenin kullanıcılarına sağladığı özellikler aşağıdaki gibidir:

- Verinin hızlı bir biçimde okunması ve entegrasyonu
- Veri ön işleme aşamasının gerçekleştirilmesi (gürültülü verilerden ayıklanması, eksik verinin tamamlanması, yeniden biçimlendirme ve ölçeklendirme)
- Özellik eliminasyonu, başarılı HPO
- Sınıflandırma görevi için klasik makine öğrenmesi algoritmalarının ve regresyon tabanlı algoritmaların kullanılması imkanı
- Kurgulanan modelin yorumlanarak, tahminlemenin tamamlanması

MLBox ardışık düzeninde 3 temel alt paket yer almaktadır. Bu paketler kendilerine atanan görevlerin otomatik olarak ilerletilmesini sağlamaktadırlar. Pre-processing (ön işleme) alt paketi; verinin okunması ve ön işleme sürecinin, Optimization (optimizasyon) alt paketi model testi ve çapraz geçerleme adımlarının, Prediction (tahminleme) alt paketi ise tahminleme sürecinin yürütülmesi için tasarlanmıştır. MLBox sadece Linux uyumlu olup, Windows ve MacOS desteği henüz bulunmamaktadır.

1.2.2. Auto-Sklearn *(Auto-Sklearn)*

Scikit-learn üzerine kurulmuş olan bir AutoML paketidir. Bu paket sayesinde algoritma seçimi ve HP ayarlaması işlemleri kullanıcı üzerinden alınarak tamamen otomatikleştirilmiştir. Auto-sklearn paketi, veri setinin nitelikleri üzerinde kategorik değişkenlerin ikili temsilenmesi (one-hot encoding), nümerik özellik standardizasyonu (numeric feature standardization), temel bileşenler analizi (principal component analysis) gibi önemli özellik çıkarımı yöntemlerini uygulamaktadır.

Auto-sklearn'deki ardışık düzen kurgusu bayezyen arama yoluyla optimize edilmektedir. Burada kullanılan bayezyen HPO sayesinde makine öğrenmesi kurgusu kontrol edilerek dengelenmektedir. Bayezyen HPO sistem içerisinde bayezyen dengeleyici (optimizer) için meta-öğrenmeyi (meta learning), optimizasyonun süreçteki devamlı kontrolü ve değerlendirmesi için de otomatik ensemble yapıyı (automated ensemble construction) kullanmaktadır.

Auto-sklearn, küçük ve orta büyüklükteki veri setlerinde iyi düzeyde performans sunmaktadır. Ancak bu paket çok büyük veri setleri üzerinde yapılan derin öğrenme sistem kurgusu için henüz uygulanabilir değildir. Paket sadece Linux tabanlı makinelerde çalışmaktadır.

1.2.3. Ağaç tabanlı ardışık düzen optimizasyon aracı *(Tree-based pipeline optimization tool-TPOT)*

TPOT; Python tabanlı bir AutoML aracıdır. Temel işlevi genetik programlamayı kullanarak makine öğrenmesinde kullanılan ardışık düzeni optimize etmektir. Scikit-learn çerçevesinin bir uzantısı şeklinde olmasına rağmen kendi regresör (regressor) ve sınıflandırıcı metotları bulunmaktadır. Diğer bir deyişle TPOT, bir AutoML ardışık düzen keşfinde olası pek çok seçenek arasından veriye en uygun olan ardışık düzeni seçtirmektedir. Bu faydasına karşın TPOT; doğal dil işleme çalışmaları için uygun değildir. Benzer şekilde verideki kategorik yapıların analizini de gerçekleştirememektedir. Böyle durumlarda TPOT kullanılması mevcut kategorik değişkenlerin nümerik hale getirilmesi ile mümkündür.

1.2.4. H2O *(H2O)*

H2O.ai şirketi tarafından hafıza tabanlı dağıtık makine öğrenmesi platformu olarak sunulmaktadır. Tamamıyla açık kaynaklıdır. R ve Python desteğini bir arada sunan sistemde istatistiksel analizler, klasik makine öğrenmesi algoritmaları ve derin öğrenme algoritmaları sunulmaktadır.

H2O içerindeki AutoML modülü kendine has bir algoritma ile ardışık düzeni inşa etmektedir. Bu ardışık düzeni, özellik mühendisliği ve HP'ler yardımıyla optimize etmektedir. Kullanıcılar H2O sayesinde, bir makine öğrenmesi iş akışının otomatikleştirilerek kullanılmasına hakim olmaktadırlar. Yani özellik seçimi, modellerin kurulması, geçerlemelerinin yapılması, seçimi ve kullanılması adımları baştan sona otomatik ilerlemektedir.

1.2.5. Auto-Keras *(Auto-Keras)*

DATA Lab tarafından açık kaynaklı bir AutoML kütüphanesi olarak üretilmiştir. Auto-Keras, Keras derin öğrenme çerçevesi üzerine geliştirilmiştir. Bu kütüphane ile en uygun mimari için otomatik arama fonksiyonları ve derin öğrenme modelleri için HPler elde edilmektedir. Auto-Keras, makine öğrenmesi sürecini basitleştirmede sinir mimari arama (neural architecture search- NAS) algoritmasını kullanmaktadır.

1.2.6. Cloud AutoML *(Cloud AutoML)*

Google tarafından hizmete sunulan Cloud AutoML ortamı, sınırlı erişim hizmetine rağmen yüksek kompleksiteye sahip modellerin kurgusunda destek sağlamaktadır. Google bu desteğini kendi transfer öğrenme ve sinir mimari arama teknolojisi üzerinden sunmaktadır.

Cloud AutoML; kullanıcının kendi verisi üzerinde model kurma sürecini grafik kullanıcı arayüzü (graphical user interface-GUI) birimi ile sağlamaktadır. Google'ın AutoML çalışmaları verinin/yapılacak işlemin türüne göre üç ana çözüm barındırmaktadır. AutoML Vision görüntü verilerin işlenmesinde, AutoML Natural Language çoklu ya da tekil etiketli metin verilerin tahminlemesinde, AutoML Translation ise çeviri modellerinin geliştirilmesinde kullanılmaktadır. Bu üç çözüm açık kaynaklı olmayıp belli sayıdaki veriler ve belli süreler için ücretsiz destek sunmaktadır. Örneğin AutoML Vision ile yapılacak bir görüntü sınıflandırmasında kullanılacak her model için 1 saat ve her ay 10 model ve 1000 görüntü üzerinde yapılacak tahminlemeler ücretsizdir.

1.3. DataRobot *(DataRobot)*

Ticari bir ürün olarak geliştirilen DataRobot, diğer AutoML araçları ve ortamlarına kıyasla oldukça kullanıcı dostu bir arayüz ile uçtan uça analiz imkanı sunmaktadır. Ayrıca diğer araçlarda çok da gündeme alınmayan, sınırlı özelliklerle sunulan görselleştirme DataRobot'ta oldukça kuvvetli bir hizmet olarak sunulmaktadır. DataRobot kendi platformu üzerinden kullanıcılarına verinin hazırlanması, özellik seçimi, algoritma seçimi, çeşitlendirilmesi, eğitilmesi, modellerin yarıştırılması ve çalışmaya hazır hale getirilmesi, çalışılan modelin farklı veri setleri üzerinden başarısının izlenmesi ve kontrol edilmesi imkanlarını sunmaktadır.

DataRobot, kendi AutoML sürecini kurgularken sürekli öğrenen uzman sistemler mantığını kullanmıştır. Yani ilk çıkış noktası insan veri bilimci, makine öğrenmesi uzmanının düşünce/karar verme biçimini modellemiş bu model üzerinden optimizasyon kurgulamıştır. Diğer ortam ve araçlardan farklı olarak DataRobot en son adıma varıldığında kullanıcıya R, Python gibi çeşitli dillerde tüm sürecin kodunu iletmektedir.

1.4. Microsoft AutoML *(Microsoft AutoML)*

Azure Machine Learning üzerinden kullanılabilen Microsoft AutoML; HPlerin seçiminde, özellik seçiminde ve model kıyaslarında üst düzey ve kendini yenileyebilen bir ardışık düzen kullanmaktadır. Analiz sürecinde insan etkisinin azaltılması felsefesiyle çalışan bu araç, klasik makine öğrenmesi algoritmalarında oldukça etkin bir başarı sunarken, çok katmanlı sinir ağlarına yönelik bir hizmeti bulunmamaktadır. Microsoft'un diğer ürünleri olan Power BI, Azure AI, Azure Cloud ile entegre biçimde çalışmaktadır. Microssoft AutoML, Eylül 2018'den beri Python entegrasyonu gerçekleştirmiştir.

2. ARAŞTIRMA *(RESEARCH)*

Bu çalışmada AutoML sürecinin faydalarını ortaya koyabilmek adına, 6 MIT (Massachusetts Institute of Technology) CSAIL (Computer Science and Artificial Intelligence Lab) doktora sonrası araştırmacı ile çalışılmıştır. Çalışma kapsamında doktora sonrası araştırmacıların seçiminde aşağıdaki kriterlere dikkat edilmiştir:

- Tezinde en az 5 klasik makine öğrenmesi algoritması ile model geliştirmiş olma
- Tez sürecinde Weka, Rapidminer gibi makine öğrenmesi araçlarını kullanmamış ancak R, Python gibi herhangi bir dili uygulamış olma
- Medikal alanda altyapısı/eğitimi olmama
- Daha önce medikal verilerle çalışmamış olma
- AutoML sürecine aşina olma

Analiz sürecinde UCI üzerinden sunulan Cervical Cancer (Risk Factors) veri setinde kullanılmıştır (http://archive.ics.uci.edu/ml/datasets/Cervical+cancer+%28Risk+Factors%29).

Sınıflandırma görevine uygun olan bu verisetinde 858 adet gözlem değeri ve 36 nitelik bulunmaktadır. 2017 yılında bağışlanan veri setinde eksik veriler de bulunmaktadır. Tahmine uygun olarak hedef nitelikler Cytology ve Biopsy'dir.

Araştırma kapsamında 2'şerli takımlar oluşturulmuş ve AutoML'in etkisini anlamak üzere iki aşamada gerçekleştirilmiştir. İlk aşamada takımlara 2 saatlik bir süre verilmiş ve Cytology hedef niteliğini tahmin etmeleri istenmiştir. Takım 1 Google AutoML, Takım 2 Auto-Keras kullanmış olup, Takım 3 herhangi bir AutoML aracı olmaksızın Python kütüphanelerinden yararlanmışladır. İkinci aşamada, sadece Takım 3'ün AutoML kullanılmasına izin verilmiş olup, diğer takımlar analiz dili bakımından serbest bırakılmıştır. Analiz sürecinde aynı veri seti, Biopsy değişkeninin tahmin edilmesi amacıyla sunulmuş olup bu sefer 1 saatlik süre tanınmıştır.

Her iki aşama sonrasında üretilen tahminleyici modellerin başarı değerleri kıyaslanmıştır. Tablo 1'de takımların modelleme-başarı değerleri verilmektedir.

Tablo 1. Modelleme-tahmin başarı sonuçları
(Accuracy results of the prediction and model)

Takımlar	Aşamalar ve Ortalama Başarı Değerleri			
	Aşama 1	Aşama 2	Aşama 1-Ortalama Başarı Değerleri (ACC)	Aşama 2-Ortalama Başarı Değerleri (ACC)
Takım 1	Var	Yok	0.78401	0.80142
Takım 2	Var	Yok	0.77933	0.75096
Takım 3	Yok	Var	0.73261	0.83422

Tablo 1 incelendiğinde AutoML araçlarının, sadece insandan oluşan takımlara göre daha performanslı sonuçlar üretebildiği görülmektedir. İnsan faktörü veri setini anlamlandırma, ön işleme sürecinde ve yöntem seçiminde daha önceki deneyimlerinden yola çıkarak, eylemler gerçekleştirmekte, modelin öğrenme sürecini dengeleyebilmek adına pek çok defa denemeler yapmaktadır. Bu denemelerde süreye bağlı kurgu ve kod hataları meydana gelebilmekte, bu hatalardan kaynaklı zaman kaybı yaşanmaktadır. Oysa AutoML süreci, bu deneyime ek olarak düzgün bir ardışık düzen kurgusu sunarak hata miktarını aza indirmektedir.

Performansın ötesinde AutoML araştırmacıya daha fazla modeli daha güçlü bir ortamda deneme imkanı sunmaktadır. Elde edilen sonuçlar [25] tarafından paylaşılan KaagleDays sonuçlarıyla benzerlik göstermektedir. KaggleDays kapsamında 8.5 saat süren 74 takımlı analiz sürecinde insan, insan ve AutoML araçlarının işbirliği kıyaslanmıştır. Yarışmada elde edilen puanlara göre insan AutoML işbirliği en iyi sonuçları elde etmiştir [25].

3. SONUÇ *(CONCLUSION)*

Makine öğrenmesi alanındaki talebin giderek arttığı bir dünyada bu alanda istenilen başarının alanın yetişmiş nitelikli insan sayısına bağlı olması önemli bir sorundur. AutoML kavramı bu soruna çözüm üretmek amacıyla geliştirilmektedir. Çünkü gerçek dünya problemi ne olursa olsun, veriyi barındıran o problemin makine öğrenmesi sürecine bir optimizasyon problemi gibi yaklaşılarak iteratif veya lineer ardışık düzen kurgusu ile cevap verilmesi önemli bir avantaj sağlamaktadır. Benzer şekilde makine öğrenmesi yaşam döngüsünde insan hatasının az ve önyargıdan daha uzak neden sonuç ilişkisi açıklanabilir, yani demokratik makine öğrenmesi sürecin yaratılması mümkün kılınabilir. Ancak AutoML ile hedeflenen insanın analiz sürecinden tamamıyla çıkarılması değildir. Daha düzgün bir iş akışı içinde insan ve makinenin birlikteliğinin sağlanmasıdır. Bu fikirden yola çıkılarak araştırma kapsamında sadece insan ve iki autoML aracı – insan işbirliğinin tahminleme görevi karşısındaki başarısına bakılmıştır. Çalışmada kullanılan araçlar, katılımcı sayısı ve tek bir veri seti üzerinden değerlendirilmesi sınırlılıklarına sahiptir. Elde edilen sonuçlara bakıldığında, daha fazla algoritma ile çalışarak daha çok modeli deneyimleyebilme ve zaman maliyeti bakımından AutoML'in makine öğrenmesi sürecinde insana önemli bir avantaj sağladığı görülmektedir.

Çeşitli raporlarda veri bilimi, makine öğrenmesindeki pek çok görevin 2020 yılına kadar makinelere devredileceği vurgusu yapılmaktadır. Halen geliştirilmekte olan AutoML bu alandaki önemli açığı kapatmakla kalmayacak, analiz süreci bakımından daha güvenilir ve demokratik bir ortam yaratacaktır.

KAYNAKLAR *(REFERENCES)*

[1] Accenture, "The Team Solution to the Data Scientist Shortage*"*, Accenture, 2013.

[2] J. G. Harris and R. Eitel-Porter, "Data scientists: 'As rare as unicorns" https://www.theguardian.com/media-network/2015/feb/12/data-scientists-as-rare-as-unicorns, 2015.

[3] insideBIGDATA, "The Data Scientist Shortage is Huge" *Here's How to Beat It,* https://insidebigdata.com/2018/12/27/data-scientist-shortage-huge-heres-beat/ , 2018.

[4] A. Woodle, *What's Driving Data Science Hiring in 2019*. https://www.datanami.com/2019/01/30/whats-driving-data-science-hiring-in-2019/ , 2019.

[5] J. R. Lloyd, D. K. Duvenaud, R. B. Grosse, J. B. Tenenbaum, Z. Ghahramani, "Automatic construction and natural-language description of nonparametric regression models". *In Proceedings of the Twenty- Eighth AAAI Conference on Artificial Intelligenc*e, 2014.

[6] M. Feurer, A. Klein, K. Eggensperger, J. Springenberg, M. Blum, F. Hutter, "Efficient and robust automated machine learning". In C. Cortes,N.D. Lawrence, D. D. Lee, M. Sugiyama,&R. Garnett (Eds.), *Advances in neural information processing systems*, Curran Associates, Inc, 2015, pp. 2962–2970.

[7] A. Romblay, "Automated Machine Learning" *Datahack Summit*, Anlytics Vidhya, 2017

[8] Bengio, Y. "Gradient-based optimization of hyperparameters", Neural computation, 2000, 12 (8), pp. 1889-1900.

[9] J. Bergstra, Y. Bengio, Y. (2012). "Random search for hyper-parameter optimization", Journal of Machine Learning Research, 2012, 13(Feb), pp.281-305.

[10] R. Kohavi ve G. John, "Automatic Parameter Selection by Minimizing Estimated Error" In: Prieditis, A., Russell, S. (eds.) Proceedings of the Twelfth International Conference on Machine Learning, Morgan Kaufmann Publishers. 1995, pp. 304–312.

[11] H. Escalante, M. Montes, E. Sucar, "Particle Swarm Model Selection". Journal of Machine Learning Research, 2009, 10, pp. 405–440.

[12] C. Thornton, F. Hutter, H. Hoos, K. Leyton-Brown, "Auto-WEKA: combined selection and hyperparameter optimization of classification algorithms". In: Dhillon, I., Koren, Y., Ghani, R., Senator, T., Bradley, P., Parekh, R., He, J., Grossman, R., Uthurusamy, R. (eds.) The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'13), ACM Press, 2013, 847–855.

[13] R. Mantovani, T. Horvath, R. Cerri, J. Vanschoren, A. Carvalho. "Hyper-Parameter Tuning of a Decision Tree Induction Algorithm". In:5th Brazilian Conference on Intelligent Systems, IEEE Computer Society Press, 2016, 37–42.

[14] S. Sanders ve C. Giraud-Carrier, "Informing the Use of Hyperparameter Optimization Through Metalearning". In: Gottumukkala, R., Ning, X., Dong, G., Raghavan, V., Aluru, S., Karypis, G., Miele, L., Wu, X. (eds.) 2017 IEEE International Conference on Big Data (Big Data). IEEE Computer Society Press, 2017.

[15] R. Olson, W. La Cava, Z. Mustahsan, A. Varik, J. Moore, "Data-driven advice for applying machine learning to bioinformatics problems". In: Proceedings of the Pacific Symposium in Biocomputing 2018, 192–203.

[16] L. Kotthoff, C. Thornton, H. Hoos, F. Hutter, K. Leyton-Brown, "Auto-WEKA 2.0: Automatic model selection and hyperparameter optimization in WEKA". Journal of Machine Learning Research, 2017, 18, 1–5.

[17] F. Hutter, R. Caruana, R. Bardenet, M. Bilenko, I. Guyon, B. Kegl, H. Larochelle, "AutoML", ICML Workshop, 2014.

[18] B. Komer, J. Bergstra, C. Eliasmith, "Hyperopt-sklearn: Automatic hyperparameter configuration for scikit-learn". In ICML workshop on AutoML, 2014.

[19] M. Feurer, A. Klein, K. Eggensperger, J. Springenberg, M. Blum, "Efficient and Robust Automated Machine Learning". Advances in Neural Information Processing Systems 2015, 28, 2962–2970.

[20] A. Balaji, A. Allen, "Benchmarking Automatic Machine Learning Frameworks", 2018, arXiv:1808.06492 CoRR abs/1808.06492. URL http://arxiv.org/abs/1808.06492.

[21] F. Hutter, H. Hoos, K. Leyton-Brown, "Sequential model-based optimization for general algorithm con_guration". In International Conference on Learning and Intelligent Optimization, Springer , 2011, 507-523.

[22] K. Swersky, J. Snoek, P. R. Adams, Freeze-thaw bayesian optimization, 2014, arXiv preprint arXiv:1406.3896.

[23] M. Munoz, L. Villanova, D. Baatar, K. Smith-Miles, "Instance spaces for machine learning classification", Machine Learning, 2018, 107(1), pp.109-147.

[24] B. Zoph, Q. V. Le, Neural architecture search with reinforcement learning, 2016, arXiv preprint arXiv:1611.01578.

[25] Y. Lu, "An end-to-end autoML aolution for tabular data at KaggleDays", Google AI Blog: 2019, May 9, https://ai.googleblog.com/2019/05/an-end-to-end-automl-solution-for.html.