



**Makale / Research Paper**

**Öğrencilerin Akademik Performanslarının Tahmin Edilmesi için  
AutoML Tekniğinin Uygulanması**

**Sevda AGHALAROVA<sup>1a</sup>, Sinem BOZKURT KESER<sup>1b\*</sup>**

<sup>1</sup>Eskişehir Osmangazi Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü,  
Eskişehir/TÜRKİYE  
[sbozkurt@ogu.edu.tr](mailto:sbozkurt@ogu.edu.tr)

**Received/Geliş:** 01.06.2021

**Accepted/Kabul:** 05.10.2021

**Öz:** Eğitsel Veri Madenciliği, çeşitli eğitim kaynaklarından elde edilen büyük miktarda verinin analizini kolaylaştırmak amacıyla veri madenciliği yöntemlerinin geliştirilmesidir. Eğitimcilerle geribildirimde bulunma, öğrencilere ders önerisinde bulunma, istenmeyen öğrenci davranışı belirleme, öğrenci performansını tahmin etme gibi konular Eğitsel Veri Madenciliği çalışma alanları arasında gösterilebilir. Doğru modeller oluşturularak bu alanlarda yapılacak iyileştirmeler ile eğitim kalitesi geliştirilebilir. Doğru modeller oluşturmak için uygun makine öğrenmesi algoritmalarının seçimi hem eğitimciler hem de veri bilimcileri için son derece önemlidir. Bu çalışmada öğrencilerin akademik performanslarını tahmin etmek amacıyla Otomatik Makine Öğrenmesi yöntemi ile çalışmada kullanılan veri seti için en iyi model araştırılmaktadır. Otomatik Makine Öğrenmesi ile veri ön işleme, model seçimi ve hiper-parametre optimizasyonu gibi zorlu görevlerle uğraşmadan en iyi model bulunabilmektedir. Çalışmada, gerçek veri seti için Dağıtılmış Rastgele Orman algoritması en iyi algoritma olarak belirlenmektedir. Izgara araması kullanılarak algoritmanın hiper-parametreleri optimize edilmektedir. Deneysel sonuçlarında, Dağıtılmış Rastgele Orman algoritmasının, varsayılan hiper-parametreleri ile doğruluk ve f-skor değerleri sırasıyla %77.50 ve %80.01 olarak elde edilmektedir. Izgara araması ile bulunan en uygun hiper-parametreler için doğruluk ve f-skor değerleri ise sırasıyla %82.30 ve %82.50 olarak hesaplanmaktadır. Çalışmada, önerilen AutoML yöntemi geleneksel makine öğrenme algoritmalarından KNN ve SVM ile karşılaştırılmaktadır. Önerilen yöntem ile her iki algoritmadan daha yüksek sonuçlar elde etmektedir.

**Anahtar Kelimeler:** Eğitsel Veri Madenciliği, Makine Öğrenmesi, Öğrenci Akademik Performans Tahmini, Otomatik Makine Öğrenmesi.

**Application of AutoML Technique for Predicting Academic  
Performance of Students**

**Abstract:** Educational Data Mining is the development of data mining methods to facilitate the analysis of large amounts of data obtained from various educational sources. Issues such as providing feedback to educators, suggesting courses to students, identifying undesirable student behavior, and predicting the academic performance of students can be shown among the fields of Educational Data Mining. The quality of education can be improved with the improvements to be made in these areas by creating the right models. The selection of suitable machine learning algorithms to build accurate models is highly important for educators and data scientists. In this study, the best model for the dataset used in the study is investigated with the Automatic Machine Learning method in order to predict the students' academic performance. The best model can be found without dealing with difficult tasks such as data preprocessing, model selection, and hyper-parameter optimization using Automatic Machine Learning. In the study, the Distributed Random Forest algorithm is determined as the best algorithm for the real-world data set. And, the hyper-parameters of the algorithm are optimized using grid search. In the results of the experiments, the default hyper-parameters of the Distributed Random Forest algorithm and the accuracy and f-score values were obtained as 77.50% and 80.01%, respectively. For the optimal hyper-parameters found by grid search, the accuracy and f-score values

*Bu makaleye atf yapmak için*

Aghalarova, S., Bozkurt Keser, S., "AutoML Tekniği Uygulayarak Öğrencilerin Akademik Performanslarının Tahmin Edilmesi" El-Cezerî Fen ve Mühendislik Dergisi 2022, 9 (2); 394-412.

*How to cite this article*

Aghalarova, S., Bozkurt Keser, S., "Predicting Students' Academic Performance Using AutoML Technique" El-Cezerî Journal of Science and Engineering, 2022, 9 (2); 394-412.

ORCID ID: <sup>a</sup>0000-0002-7322-9477; <sup>b</sup>0000-0002-8013-6922

are calculated as 82.30% and 82.50%, respectively. In the study, the proposed AutoML method is compared with traditional machine learning algorithms include KNN and SVM. The proposed method achieves higher results than both algorithms.

**Keywords:** Educational Data Mining, Machine Learning, Prediction Student Academic Performance, AutoML.

## 1. Giriş

Toplumdaki kültürlenme süreci olan eğitim sosyal, politik, ekonomik ve kültürel amaçların gerçekleştirilmesinde önemli rol oynamaktadır. Eğitim, birey ve toplum açısından değerlendirilebilir. Eğitim, birey bakımından öğrenme ve kalkınmayı ifade ederken toplumsal açıdan ise eğitim ihtiyacının karşılanmasını sağlayan, toplumun yaratıcılık yönünü ve verimliliğini artıran bir birim olarak açılabilir [1]. Eğitim, insanların hayata ilişkin fikirlerinin oluşmasında önemli bir rol oynayarak tüm toplumun bilincini geliştirebilir. Teknolojinin eğitim ve öğretim sistemine entegrasyonu öğretme ve öğrenme becerilerini etkilemektedir. Teknoloji sayesinde, web tabanlı eğitim sistemleri katlanarak artmaktadır. E-öğrenme kaynaklarının artması, birden çok kaynaktan gelen veriyi farklı formatlarda depolamaya yönelmektedir [2]. Teknoloji ile birlikte okullarda çok daha verimli ve etkili sınıf ortamları oluşmaktadır. Ayrıca, öğrenimi iyileştirmek için çevrimiçi öğrenme ortamlarından elde edilebilen verilere erişilebilir [3]. Bu veriler görüntüler, ses, video ve birçok farklı veri formatında olmakta ve geniş hacimli veri tabanlarında tutulmaktadır [4].

Eğitsel veri madenciliği, öğrenci verilerinden gizli bilgileri çıkarmak için modeller geliştirmeye odaklanan, öğrencilerin akademik performanslarının geliştirilebileceği veri madenciliği alanlarından biridir [5]. EVM uygulanarak elde edilen bilgiler ile yüksek öğretim kurumlarındaki akademik öğretim programlarını belirlemek için planlamacılara destek sağlanabilir, yeni müfredat teklifleri geliştirilebilir, başarısızlık oranları düşürülebilir ve öğrenci davranışı daha iyi analiz edilebilir [6]. EVM'nin sağladığı katkı, verilerin analizi ve görselleştirilmesi, öğrenci performansını tahmin etme, öğrenci modelleme, istenmeyen öğrenci davranışlarını tespit etme, öğrencileri gruplama, eğitim yazılımı oluşturma, e-öğrenme sistemlerinin formülasyonu, eğitim verilerinin kümelenebilmesi gibi çeşitli kategorilere ayrılabilir [5]. Sınıflandırma, regresyon ve kümeleme eğitim verilerinde uygulanan birkaç popüler EVM yöntemlerindedir [7]. EVM süreci birkaç adımdan oluşmaktadır. İlk olarak yapılan çalışmanın temel amacı belirlenmektedir. Daha sonra veriler farklı eğitim ortamlarından çıkarılmaktadır. Bir sonraki adım veri ön işleme adımından oluşmaktadır. Veri ön işleme adımında veriler temizlenmekte, dönüştürülmekte ve aktarılmaktadır. Sonrasında EVM yöntemleri uygulanmaktadır. Son olarak, oluşturulan modeller, eğitim verilerinden bilgi çıkarmak için yorumlanmaktadır [8].

Öğrenci akademik performansının doğru tahmini, akademik kaliteyi artırmanın ve daha iyi eğitim hizmetleri sağlamanın bir yoludur. Bu çalışmada, öğrenci akademik performansını etkileyen öznitelikleri içeren gerçek bir veri seti kullanılmaktadır. Çalışmanın amacı, Otomatik Makine Öğrenmesi (AutoML) yöntemini gerçek eğitim veri seti üzerinde uygulayarak en uygun modeli belirlemektir. AutoML yöntemi ile ön işleme aşamasında ve en uygun modelin seçiminde kullanıcı müdahalesine ihtiyaç duyulmadan gerekli adımlar otomatik olarak uygulanarak üzerinde çalışılan veri seti için Dağıtılmış Rastgele Orman (*Distributed Random Forest*, DRF) algoritması en uygun algoritma olarak belirlenmektedir. DRF algoritması için AutoML içerisinde hiper-parametre optimizasyonu yapılmadığından ayrıca DRF algoritması için ızgara araması ile en uygun hiper-parametreler belirlenmektedir. Deney sonuçlarında, DRF algoritmasının AutoML için tanımlanan varsayılan hiper-parametreleri ile doğruluk ve f-skor değerleri sırasıyla %77.50 ve %80.01 olarak elde edilmiştir. Izgara araması kullanılarak bulunan optimal hiper-parametreler ile doğruluk ve f-skor değerleri ise sırasıyla %82.30 ve %82.50 olarak hesaplanmaktadır. Önerilen yöntem ile geleneksel makine öğrenme yöntemlerinden K-En Yakın Komşu (*K-Nearest Neighbors*, KNN) ve Destek Vektör Makineleri (*Support Vector Machine*, SVM) algoritmaları karşılaştırılmaktadır. KNN algoritmasında doğruluk ve f-skor değerleri sırasıyla %61.1 ve %61.01, SVM algoritmasında

ise sırasıyla %64.5 ve %64.4 olarak elde edilmektedir. Elde edilen sonuçlar önerilen yöntemin başarısını ve etkinliğini kanıtlamaktadır.

## 2. Literatür Özeti

Öğrencilerin akademik performanslarının, sınav sonuçlarının, risk durumlarının tahmin edilmesi EVM alanında yaygın olarak araştırılan konulardır. Bu bölümde farklı makine öğrenmesi algoritmalarını kullanarak öğrencilerinin performansını tahmin etmeye yönelik literatürde yapılan çalışmalar incelenmektedir. Bu çalışmalar Tablo 1’de özetlenmektedir.

Ortaokul öğrencilerinin akademik performansının tahmin edilmesi için eğitsel ortamlardan toplanan veri setleri üzerinde literatürde çeşitli makine öğrenme yöntemleri uygulanarak analizler gerçekleştirilmiştir [9]. Ghorbani ve Ghousi, RF, KNN, ANN, XGBoost, SVM, DT, LR ve NB sınıflandırma algoritmalarını uygulamakta ve en iyi doğruluk değeri RF algoritması ile bulunmaktadır [10]. Kaunang ve Rotikan öğrencinin akademik performans tahmin modelini oluşturmak için DT ve RF algoritmalarını kullanmışlardır ve en iyi doğruluk değerini DT algoritması ile bulmuşlardır [11]. Asif ve ark sadece üniversite öncesi ve üniversitedeki ilk yıllara ait sınav notlarını kullanarak öğrencilerin üniversite son sınıftaki mezuniyet performanslarını tahmin etmek için RF, DT, NB ve NN algoritmalarını kullanmaktadırlar. En iyi performans değerleri NB algoritmasında gözlemlenmektedir [12]. Quinn ve Gray bir öğrencinin dersi geçip geçmeyeceğini tahmin etmek amacıyla öğrencilerin altı haftalık ve on haftalık verileri üzerinde RF, GBM, Doğrusal diskriminant analizi (*Linear discriminant analysis*), KNN algoritmalarını uygulamaktadırlar. En iyi doğruluk değerlerine, on haftalık veri seti için RF algoritması ile ulaşmışlardır [13]. Gandy ve ark üniversitenin ilk iki yılı boyunca öğrencilerin akademik performansını tahmin etmek amacıyla karar ağacı modelini uygulamaktadırlar. Çalışmada öğrenci bilgi sisteminden alınan verilerden faydalanılmaktadır. Uygulanan modeller çok benzer sonuçlar göstermektedir. XGBoost modelinde diğer modellere kıyasla daha iyi sonuçlar elde edilmektedir [14]. Ahammad ve ark öğrencilerin ortaokul sertifika sınavı sonuçlarına ve akademik performanslarına göre bir model geliştirmektedirler. Öğrencilerin sınav sonuçlarını tahmin etmek amacıyla NB, KNN, SVM, XGBoost ve MLP algoritmalarını kullanmaktadırlar. Deney sonuçlarında, XGBoost algoritması diğer algoritmalarından daha iyi performans göstermektedir [15]. Protikuzzaman ve ark Bangladeş'teki Bangabandhu Şeyh Mujibur Rahman Bilim ve Teknoloji Üniversitesi'ndeki (BSMRSTU) öğrencilerin giriş sınavına katıldıktan sonra veya katılmadan önce üniversiteye kabul edilme şansını öngörmek amacıyla LightGBM, XGBoost ve GBM algoritmalarını uygulamaktadırlar. Sonuçlar, modellerin yüksek doğrulukta öngördüğünü ve diğer fakülte ve üniversitelere kabul durumunu tahmin etmek için uygulanabileceğini göstermektedir [16]. Wandera ve ark Afrika ülkelerinde okul performansını etkileyebilecek önemli özellikleri belirlemek için EVM yaklaşımlarını ve ağaç-tabanlı teknikleri uygulamaya odaklanmaktadırlar. LightGBM, geliştirilen tahmin modelinin temelini oluşturmaktadır. LightGBM algoritması yüksek doğruluk ile XGBoost, DT ve RF algoritmalarından daha iyi performans göstermektedir [17]. Sagar ve ark öğrencilerin programlama performansını tahmin etmek amacıyla GLM, XGBoost, RF ve GBM algoritmalarını uygulamaktadırlar. Bunun için *HackerEarth* web sitesi ve üniversite tabanlı bir programlama ortamı olan kurum içi programlama portalı olmakla iki farklı ortamdan elde edilen veriler kullanılmaktadır. Her iki veri seti üzerinde XGBoost algoritması daha iyi sonuçlar sergilemektedir [18]. Alshabandar ve ark çalışmasında öğrencilerin akademik performanslarını tahmin etmek amacıyla sınıflandırma ve regresyon analizi uygulamaktadırlar. Her iki yöntem için öğrenci veri setleri üzerinde RF, MLP, ANN, GBM ve GLM algoritmaları uygulanmaktadır. RF en düşük Kareler Ortalamasının Karekökü (RMSE, *Root Mean Square Error*) değerine sahipken, GBM algoritması ile öğrenci performansında en yüksek doğruluk değeri elde edilmektedir [19]. Liu ve ark, öğrenim yönetim sisteminden (*learning management system*) gelen verilerden ve sınıf ortamından elde edilen videolardan yararlanarak öğrencilerin akademik performanslarını tahmin etmeye çalışmaktadır.

**Tablo 1.** EVM alanında literatürde yapılan çalışmalar

Yazar	Amaç	Kullanılan Algoritmalar	En iyi Sonuç
Ghorbani ve Ghousi (2020)	Akademik performansı tahmin etme	Rastgele Orman (Random Forest, RF), KNN, Yapay Sinir Ağları ( <i>Artificial Neural Network</i> , ANN), Aşırı Gradyan Arttırma ( <i>Extreme Gradient Boosting</i> , XGBoost), SVM, Karar ağacı ( <i>Decision Tree</i> , DT), Lojistik Regresyon ( <i>Logistic Regression</i> , LR), Naif Bayes ( <i>Naive Bayes</i> , NB),	RF
Kaunang ve Rotikan (2018)	Akademik performansı tahmin etme	DT, RF	DT
Asif ve ark (2017)	Akademik performansı tahmin etme	RF, DT, NB, NN	NB
Quinn ve Gray (2020)	Akademik performansı tahmin etme	RF, Gradyan Arttırma makineleri ( <i>Gradient Boosting Machine</i> , GBM), Doğrusal diskriminant analizi (Linear discriminant analysis, LDA), KNN	RF
Gandy ve ark (2019)	Akademik performansı tahmin etme	XGBoost, ANN	XGBoost
Ahammad ve ark (2021)	Öğrencilerin sınav sonuçlarını tahmin etme	NB, KNN, SVM, XGBoost, Çok Katmanlı Algılayıcı ( <i>Multilayer Perceptron</i> , MLP)	MLP
Ghorbani ve Ghousi (2020)	Öğrenci performansını tahmin etme	RF, KNN, ANN, XGBoost, SVM, DT, LR, NB	RF
Protikuzzaman ve ark (2020)	Öğrencilerin üniversiteye kabul edilme fırsatını tahmin etme	Hafif Gradyan Arttırma Makineleri ( <i>Light Gradient Boosting Machine</i> , LightGBM), XGBoost, GBM	GBM
Wandera ve ark (2019)	Akademik performansı tahmin etme	LightGBM, XGBoost, DT, RF	LightGBM
Sagar ve ark (2016)	Öğrencilerin programlama performansını tahmin etme	Genelleştirilmiş Doğrusal Modeller ( <i>General linear model</i> , GLM), XGBoost, RF, GBM	XGBoost
Alshabandar ve ark (2020)	Akademik performansı tahmin etme	GLM, RF, MLP, ANN, GBM	GBM, RF
Liu ve ark (2020)	Akademik performansı tahmin etme	DT, NB, LR, RF	LR, RF
Ha ve ark (2020)	Akademik performansı tahmin etme	NB, RF, MLP, SVM, Rastgele ağaçlar, Part, OneR	MLP, NB
Pekuwali (2020)	Akademik performansı tahmin etme	NB	NB
Asril ve Isa (2020)	Öğrenci çalışma sürelerini tahmin etme	KNN	KNN
Bunkar ve Tanwani (2020)	Öğrenci davranış analizi ve Akademik performansı tahmin etme	C4.5, CART	C4.5
Tomasevic ve ark (2020)	Akademik performansı tahmin etme	KNN, SVM, ANN, DT, NB, LR, Bayesian Regresyon, Doğrusal Regresyon	ANN
Farissi ve ark (2020)	Akademik performansı tahmin etme	ANN, DT, RF, Torbalama, Güçlendirme, Oylama	RF
Abbasoğlu Banu (2020)	Akademik performansı tahmin etme	LR, Doğrusal SVM, Doğrusal Olmayan SVM, RF, NB, Torbalama, KNN, MLP	LR
Sathe ve Adamuthe (2021)	Akademik performansı tahmin etme	C5.0, J48, CART, NB, KNN, RF, SVM	C5.0, RF
Tsiakmaki ve ark (2020)	Akademik performansı tahmin etme	AutoML	AutoML
Drăgulescu ve Bucos (2020)	Akademik performansı tahmin etme	DT, RF, Ekstra Ağaç, AutoML	AutoML
Tsiakmaki ve ark (2021)	Akademik performansı tahmin etme	Bulanık ortalama toplama akıl yürütme kuralı ( <i>Fuzzy mean aggregation reasoning rule</i> , FMARR), Örüntü ağaçları ( <i>The Pattern trees</i> , PT), Yukarıdan aşağıya örüntü ağaçları ( <i>Top-down pattern trees</i> , PTTD), Değiştirilmiş Bulanık Model Sınıflandırıcı ( <i>Modified Fuzzy Pattern Classifier</i> , MFPC), Yerel ve Küresel Genetik FPC'ler ( <i>Local and Global Genetic FPCs</i> , FPC GA/LGA), NB, RF, NN, Arttırma ( <i>Boosting</i> )	PT, PTTD

Bu çalışmada, öğrencilerin sınıftaki davranışlarının akademik performanslarına etkisi de araştırılmaktadır. Elde edilen veriler üzerinde DT, NB, LR ve RF algoritmaları ile öğrencilerin akademik performansları tahmin edilmektedir. Tüm algoritmalar birbirine yakın sonuçlar vermektedir [20]. Ha ve ark üniversite öğrencilerinden elde edilen verileri kullanarak öğrencilerin son notlarını tahmin etmeyi amaçlamaktadır. Çalışmada NB, SVM, MLP, RF, Rastgele ağaçlar, PART ve OneR algoritmaları kullanılmaktadır. NB ve MLP algoritmaları diğer algoritmalara göre daha iyi sonuçlar göstermektedir [21]. Pekuwali çalışmasında bir önceki dönem öğrenci final notu verilerine dayanarak gelecek dönemdeki final notlarının tahminini incelemektedir. Bu amaçla NB sınıflandırma algoritmasını kullanmaktadır. Sonuç olarak bu algoritmanın yüksek doğrulukta sınıflandırma yaptığı görülmektedir [6]. Asril ve Isa öğrencilerden toplanan veriler üzerinde öğrencilerin son sınıf başarılarına göre onların çalışma sürelerini tahmin etmeye yönelik çalışma gerçekleştirmektedirler. Bu çalışmada, KNN algoritması kullanılmaktadır. Sonuç olarak algoritmanın öğrenci çalışma süresini tahmin etmek ve öğrencilerin mezuniyet oranlarını artırmak için uygulanabilir olacağı gözlemlenmektedir [22]. Bunkar ve Tanwani öğrencilerin öğrenme metodolojisini analiz etmek amacıyla kümeleme, başarısının tahmini için ise sınıflandırma yöntemini kullanmaktadır. Öğrencilerin akademik performanslarını tahmin etmek amacıyla C4.5 ve CART karar ağacı sınıflandırma algoritmaları kullanılmaktadır. C4.5 karar ağacı algoritması ile daha az bellek kullanımını ve daha kısa çalışma zamanında daha yüksek doğrulukta sonuçlar elde edilmektedir [23]. Tomasevic ve ark öğrencilerin akademik performanslarını tahmin etmek amacıyla, geçmiş öğrenci performansı, öğrenci katılımı ve öğrenci demografik verileri üzerinde farklı sınıflandırma ve regresyon algoritmaları uygulamaktadırlar. Hem sınıflandırma, hem de regresyon analizinde en iyi başarı ANN algoritması ile öğrenci katılım ve geçmiş öğrenci performansı verilerini kullanarak elde edilmektedir [3]. Farissi ve ark öğrencilerden toplanan veriler üzerinde Genetik algoritma özellik seçme yöntemini uygulayarak öğrencilerin performanslarını tahmin etmek amacıyla farklı sınıflandırma algoritmalarını kullanmaktadırlar. Sonuç olarak RF algoritması ile en iyi sonuçlar elde edilmektedir [24]. Abbasoğlu öğrencilerden toplanan demografik ve sosyoekonomik verilerin, öğrencilerin başarı ortalamalarına olan etkilerini analiz etmektedir. Bu amaçla LR, Doğrusal SVM, Doğrusal Olmayan SVM, RF, NB, Torbalama, KNN ve MLP algoritmaları kullanılmaktadır. Deney sonuçlarında, LR ile en iyi sonuçlar elde edilmektedir [25]. Sathe ve Adamuthe öğrencilerin performansının tahmini için üç farklı veri seti üzerinde C5.0, J48, CART, NB, KNN, RF ve SVM algoritmalarını uygulamaktadırlar. Aynı zamanda, bu algoritmaların farklı parametre değerleri dikkate alınarak doğruluk değerleri karşılaştırılmaktadır. Deney sonuçlarında, RF ve C5.0 algoritmalarının diğer algoritmalara göre daha iyi sonuçlar verdiği gözlemlenmektedir [26].

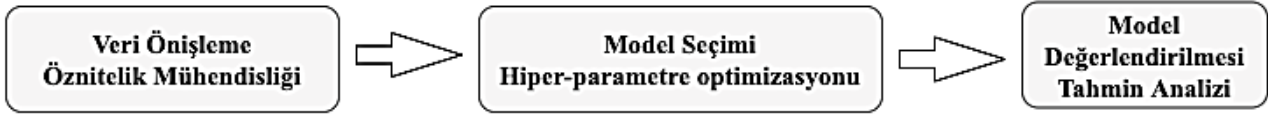
Literatürde öğrencilerin akademik performansının tahmin edilmesi üzerine yapılan çalışmalar incelendiğinde makine öğrenmesi algoritmaları kullanılarak yapılan çalışmaların çok sayıda olduğu, AutoML yöntemi kullanılarak yapılan çalışmaların ise kısıtlı olduğu görülmektedir [27, 28, 29]. Tsiakmaki ve ark desteklenen üç zorunlu derste öğrencilerin performansının erken belirlenmesi için öğrencilerin Moodle e-öğrenme platformu tarafından elde edilen verileri üzerinde AutoML tekniğini kullanmaktadırlar. Şeffaf ve yorumlanabilir sonuçlar elde etmek amacıyla arama alanı sadece ağaç-tabanlı ve kural-tabanlı modellerle sınırlandırılmaktadır. Deney sonuçlarında, AutoML tekniği ile tutarlı şekilde üstün sonuçlar elde edildiği kanıtlanmaktadır [27]. Drăgulescu ve Bucos öğrencilerin başarılarını tahmin etmeğe yönelik model geliştirmeyi amaçlamaktadırlar. Deneylerde veri temizleme, ön işleme, özellik mühendisliği işlemleri standart yöntemlerle yapılmaktadır. Çalışmanın amacı, hiper-parametre optimizasyonunu AutoML yöntemleri ile gerçekleştirilerek model performansının iyileştirilmeye çalışılmasıdır. Çalışmada DT, Ekstra Ağaç ve RF sınıflandırıcıları kullanılmaktadır. Deneysel sonuçlar, AutoML yöntemi ile eğitilen modellerin daha iyi performans gösterdiğini kanıtlamaktadır [28]. Tsiakmaki ve ark diğer bir çalışmasında ise öğrencilerin akademik performansını tahmin etmek amacıyla AutoML yöntemlerini birleştiren bulanık tabanlı aktif öğrenme yöntemini tanıtmaktadırlar. Çalışmada bulanık tabanlı algoritmalar ile

sınıflandırma yapılmakta ve bu algoritmaların hiper-parametrelerinin AutoML ile optimizasyonu gerçekleştirilmektedir. Önerilen bulanık tabanlı yöntemin yükseköğretimdeki düşük performanslı öğrencilerin tespit edilmesinde önemli bir araç olarak hizmet edebileceği kanıtlanmaktadır. Deney sonuçları, önerilen yöntemin akademik performansı iyileştirmek için öğrencilere destek olabileceğini kanıtlamaktadır [29].

### 3. Materyal ve Yöntem

#### 3.1. Otomatik Makine Öğrenmesi (AutoML)

AutoML, bir veri seti üzerinde optimum performansı elde edebilmek için makine öğrenmesinin ardışık düzeninin otomatik olarak inşa edilmesini sağlamaktadır. AutoML yönteminin vaat ettiği şey, veri ön işleme, model seçimi ve hiper-parametre optimizasyonu gibi zor ve zaman alıcı görevleri otomatikleştirebilmesi ve çok az insan çabasıyla iyi ardışık düzenler oluşturabilmesidir [30]. Hesaplama gücünün katlanarak artmasıyla, AutoML hem endüstride hem de akademide son dönemlerde uygulanan bir araç haline gelmektedir [31]. AutoML araçları, çok sayıda makine öğrenmesi modelini eğitmek için basit bir arayüz sağlamaktadır. Bunun sonucunda yeni veya ileri düzey uygulayıcılar için yararlı bir araç olmaktadır [32]. AutoML sisteminin desteklediği görevler, sınıflandırma ve regresyon olmaktadır [30]. Şekil 1 ile AutoML ardışık düzeni gösterilmektedir.

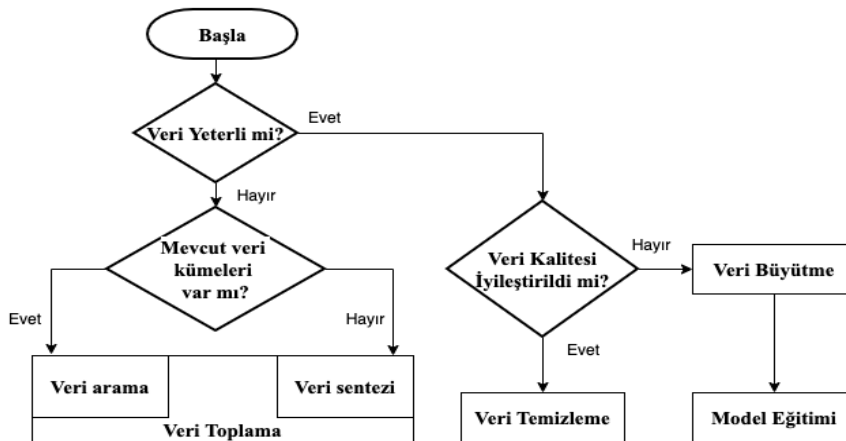


Şekil 1. AutoML iş akış şeması

AutoML aracı, üç aşamalı ortak bir ardışık düzeni izlemektedir (Şekil 1). Genel olarak, bu üç bileşen, en iyi sonucu elde etmek için yinelemeli olarak optimize edilmektedir [33]. AutoML ardışık düzeni veri ön işleme, öznitelik mühendisliği, model oluşturma, hiper-parametre optimizasyonu ve model değerlendirme olarak tanımlanan birkaç işlemden oluşmaktadır.

#### 3.1.1. Veri Ön işleme

Veri ön işleme, AutoML ardışık düzeni için tanımlanan ilk adımdır. Bu adımda aykırı değerler, eksik veriler, ölçeklendirme, gürültü filtreleme gibi temel işlemler gerçekleştirilmektedir. Şekil 2 ile veri toplama, veri temizleme ve veri artırma alt adımlarını içeren veri ön işleme iş akışı verilmektedir [31].



Şekil 2. Veri hazırlama için akış şeması

Veri toplama aşaması, yeni bir veri seti oluşturma veya mevcut veri setini genişletmek için gereken bir adımdır. Veri yeterli değilse ya yeni veri aranmakta ya da veri simülatörü ile yeni veri sentezlenmektedir. Veri toplama aşamasında verinin amaca uygun olması gerekmektedir [31].

Veri temizleme adımında ise, daha önceki aşamalardan elde edilen veri ile ilgili gürültülü veriler tespit edilmekte ve temizlenmektedir. Başka bir deyişle, verilerin EVM yöntemlerinde kullanılmaya uygun hale getirilmesidir. Gürültülü verilerin temizlenmesi, veri kalitesini iyileştirmekte ve modelin performansını artırmaktadır [34]. Veri temizleme adımları, AutoML yönteminde genellikle sabit kodlanmaktadır. Bu sabit veri temizleme adımları; eksik değerlerin tespit edilmesi, aykırı değerler gibi yanlış değerlere sahip örneklerin çıkarılması ve özelliklerin normalleştirilmiş bir aralıkta ölçeklendirilmesini içermektedir [34]. Veri temizleme görevi, AutoML araçları tarafından çok iyi yerine getirilemediği için önemli ölçüde insan müdahalesi gerektirmektedir [33].

Veri büyütme, model performansının iyileştirilmesinde önemli bir rol oynamaktadır. Bununla birlikte veri büyütme, model eğitiminin gereğinden fazla ezberlemesini önlemek için bir düzenleyici görevi görmektedir [31].

### 3.1.2. Öznitelik Mühendisliği

Öznitelik mühendisliği, algoritmalar ve modeller tarafından kullanılmak üzere ham verilerden öznitelikler çıkarma ve seçme sürecidir. Belirli bir modelin performansı, özniteliklerinin kalitesine büyük ölçüde bağlı olduğundan bu adım AutoML ardışık düzeni için oldukça önemlidir [35]. Öznitelik mühendisliği öznitelik çıkarma, öznitelik oluşturma ve öznitelik seçimi olmak üzere üç alt göreve ayrılmaktadır. AutoML kapsamında öznitelik çıkarma ve öznitelik oluşturma genellikle öznitelik üretimi olarak tanımlanmaktadır [31, 34]. Öznitelik oluşturma, modelin performansını geliştirmek amacıyla temel öznitelik uzayından veya ham verilerden elde edilen orijinal öznitelikler arasındaki eksik ilişkileri keşfederek yeni öznitelikler oluşturan bir süreçtir [31, 34]. Veri setine dayalı olarak, bir dizi aday öznitelik oluşturulmakta ve derecelendirilmektedir. Yüksek dereceli öznitelikler değerlendirilmekte ve veri setine eklenmektedir. Bu etaplar birkaç kez yinelenmektedir. Yeni öznitelikler, orijinal öznitelikleri dönüştüren önceden tanımlanmış bir dizi operatör kullanılarak oluşturulmaktadır [34].

Öznitelik çıkarma, belirli işlevleri uygulayarak gerçekleştirilen bir boyut azaltma işlemidir. Öznitelik seçiminden farklı olarak, öznitelik çıkarma, orijinal öznitelikleri değiştirmektedir. En yaygın olarak, temel bileşen analizi (*Principal Component Analysis*, PCA), bağımsız bileşen analizi, izomap, doğrusal olmayan boyutluluk azaltma ve doğrusal diskriminant analizi (*Linear Discriminant Analysis*, LDA) yaklaşımları uygulanmaktadır [31].

Öznitelik seçimi, modelin eğitim sürecini hızlandırmak amacıyla gereksiz öznitelikleri azaltarak orijinal öznitelik kümesinin bir alt kümesini oluşturmaktadır [34]. Bu işlem sayesinde aşırı öğrenmeden (*overfitting*) kaçınılarak model performansı iyileştirilmektedir. Seçilen öznitelikler genellikle farklı olup sınıf değerleriyle oldukça ilişkilidir [31].

### 3.1.3. Model Seçimi

Öznitelikler elde edildikten sonra, sınıf etiketlerini tahmin etmek amacıyla bir model bulmak gerekmektedir. Model seçim adımı, her biri farklı hiper-parametre setine sahip birden çok farklı model türünü eğitmek amacıyla gerçekleştirilmektedir. Ardından ise en iyi model veya model grubu nihai model olarak seçilmektedir [33]. Model seçimi iki bileşen içermektedir. İlk olarak bazı sınıflandırıcılar seçilmekte ve sonrasında bu sınıflandırıcıların hiper-parametreleri ayarlanmaktadır [35]. AutoML yönteminin mevcut olan farklı araçları, model oluşturmak için makine öğrenmesi algoritmalarının bir koleksiyonunu desteklemektedir. Bunlar LR, Ağaç tabanlı algoritmalar, SVM ve ANN modellerini içermekte olup ancak bunlarla sınırlı kalmamaktadır.

AutoML sürecinin başarısı daha çok hiper-parametrelere bağlıdır. Hiper-parametre optimizasyonu için AutoML yönteminin kullandığı en popüler yöntemlerden bazıları ızgara araması, rastgele arama ve Bayes aramadır [33]. Izgara araması, arama alanında tüm noktaları değerlendirdikten sonra en iyi performans gösteren noktayı seçmektedir. Izgara araması basittir ve paralel uygulamayı desteklemektedir, ancak, hiper-parametrelerin boyutluluğuyla birlikte deneme sayısı katlanarak arttığından, hiper-parametre alanı çok büyük olduğunda hesaplama açısından pahalı ve verimsiz olmaktadır. Rastgele arama, rastgele seçilmiş bir dizi noktadan en iyi noktayı seçmektedir [33]. Bayes optimizasyonu ise ünlü Bayes teoremini kullanmaktadır. Bayes optimizasyonu algoritmasında birçok olasılık modeli kullanılabilir, ancak Gauss Süreci çoğunlukla tercih edilen yöntemdir [31].

### 3.1.4. Performans Değerlendirmesi

Modelin performansının değerlendirilmesinde kullanılan  $n \times n$  boyutuna sahip karmaşıklık matrisi gerçek ve tahmin edilen sınıflandırmayı göstermektedir. Burada  $n$ , farklı sınıfların sayısını ifade etmektedir. İkili sınıflandırma problemi ele alınırsa  $n=2$  olacaktır [36]. Doğruluk, kesinlik, hatırlama, f-skor ve Eğri Altındaki Alan (*Area Under the Curve*, AUC) gibi farklı performans ölçütleri karmaşıklık matrisinde yer alan gerçek pozitif, yanlış pozitif, gerçek negatif ve yanlış negatif değerler kullanılarak hesaplanmaktadır. Performans ölçütleri, sınıflandırıcının nasıl performans gösterdiğini anlamaya yardımcı olmaktadır. Tablo 2’de öğrencilerin akademik performansına ilişkin  $2 \times 2$  boyutuna sahip karmaşıklık matrisi verilmektedir.

**Tablo 2.** Karmaşıklık Matrisi.

		Gerçek	
		Geçti	Kaldı
Tahmini	Geçti	Doğru Pozitif (DP)	Yanlış Pozitif (YP)
	Kaldı	Yanlış Negatif (YN)	Doğru Negatif (DN)

- Doğru Pozitif (DP): Öğrencinin dersten geçme durumunu ifade eder.
- Doğru Negatif (DN): Öğrencinin dersten kalma durumunu ifade eder.
- Yanlış Pozitif (YP): Dersten kalan öğrencinin yanlış bir şekilde geçti olarak belirlenmesidir.
- Yanlış Negatif (YN): Dersten geçen öğrencinin yanlış bir şekilde kaldı olarak belirlenmesidir.

Yukarıda bahsedilen doğruluk, kesinlik, hatırlama, f-skor ve AUC performans ölçütleri aşağıdaki şekilde hesaplanmaktadır.

Doğruluk: Doğru olan tahmin sayısının toplam tahmin sayısına olan oranıdır.

$$\text{Doğruluk} = \frac{DP + DN}{DP + DN + YP + YN} \quad (1)$$

Kesinlik: Doğru tahmin edilen pozitif sınıfların sayısını ölçmektedir.

$$\text{Kesinlik} = \frac{DP}{DP + YP} \quad (2)$$



Hatırlama: Algoritmanın pozitif örnekleri doğru tahmin etme gücünü ifade etmektedir.

$$\text{Hatırlama} = \frac{DP}{DP + YN} \quad (3)$$

F-skor: Kesinlik ve hatırlama ölçütlerinin geometrik ortalamasıdır.

$$F - \text{skor} = \frac{2 * \text{kesinlik} * \text{hatırlama}}{\text{kesinlik} + \text{hatırlama}} \quad (4)$$

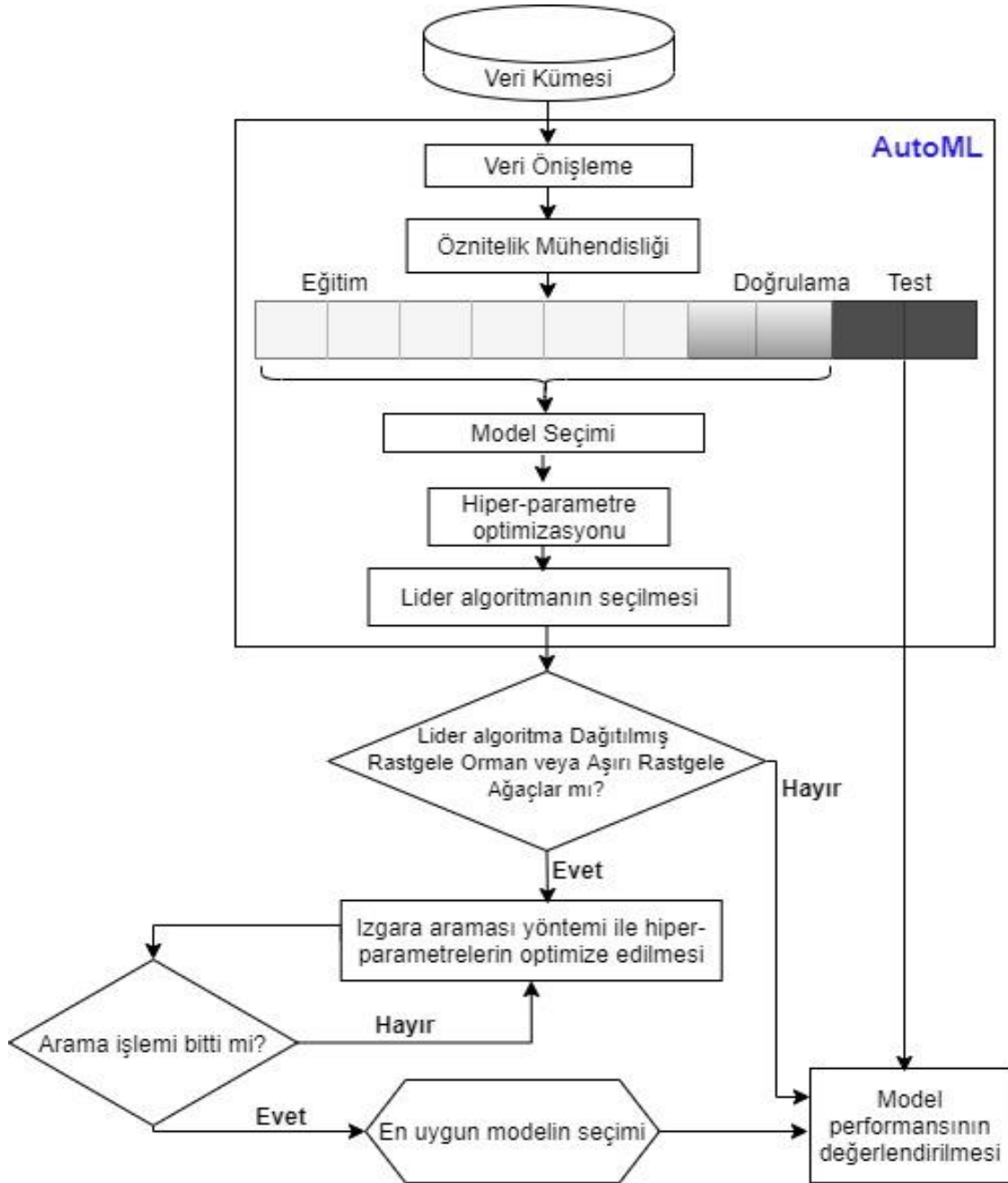
Burada, AUC: Alıcı Çalışma Karakteristiği (*Receiver Operating Characteristic*, ROC) farklı sınıflar için bir olasılık eğrisidir. ROC eğrisinde x-ekseninde Yanlış Pozitif Oran (*False Positive Rate*, YPO) ve y-ekseninde Gerçek Pozitif Oran (*True Positive Rate*, DPO) vardır. AUC ise ROC eğrisi altında kalan alanı ifade etmektedir [37].

### 3.2. Önerilen Yöntem

Bu çalışmada öğrencilerin akademik performanslarını tahmin etmek amacıyla AutoML yöntemi uygulanmaktadır. Literatürde farklı AutoKeras, MLBox, TPOT, H2O vb. gibi çeşitli AutoML araçları kullanılmaktadır. Bu çalışmada ise H2O AutoML aracı kullanılarak öğrencilerin akademik performansları tahmin edilmektedir. H2O aracı, H2O çerçevesine dahil edilen, kullanımı kolay, ölçeklenebilir, tamamen açık kaynaklı, kurumsal bir ortamda kullanıma uygun yüksek kaliteli modeller üreten, dağıtılmış makine öğrenmesi platformudur [32]. H2O AutoML aracı, kullanıcı tarafından belirlenen bir zaman sınırı dahilinde birçok modelin otomatik eğitimini ve ayarlanmasını içeren makine öğrenmesi iş akışını otomatikleştirmek için kullanılabilir [32]. H2O AutoML, tüm H2O denetimli öğrenme algoritmaları için otomatik veri ön işleme sağlamaktadır. Güncel versiyonda desteklenen tek seçenek *preprocessing = ["target\_encoding"]*: Hedef Kodlayıcı modelini otomatik olarak ayarlamakta ve ağaç tabanlı algoritmalar (XGBoost, GBM ve RF) için sütunlara uygulanmaktadır. XGBoost modelleri için otomatik atama, normalleştirme ve tek-sıcak kodlamayı (*one-hot encoding*) içermektedir. Aynı zamanda ağaç tabanlı modellerde kategorik veriler işlenebilmektedir. Otomatik metin kodlama gibi ek veri ön işleme adımlarının yanı sıra otomatik boyutluluk azaltma için öznelik seçimi ve öznelik çıkarma, H2O AutoML yol haritasının bir parçasıdır [32]. H2O GLM, DRF, XGBoost, GBM, Derin Öğrenme, Aşırı Rasgele Ağaçlar ve yığılma topluluk algoritmalarını desteklemektedir [30]. Hiper-parametre optimizasyonu aşamasında H2O, ızgara araması kullanmaktadır. H2O, geleneksel ızgara araması ve rastgele ızgara araması yöntemlerini uygulamaktadır. H2O çok sınırlı bir GPU desteği sunmaktadır. Yalnızca XGBoost modelleri GPU ile eğitilebilir, diğer modellerin eğitimi CPU desteği ile sınırlıdır [30]. Şekil 3 ile bu çalışmada önerilen yönteme ait akış diyagramı verilmektedir.

İlk aşamada eğitsel ortamdan elde edilerek hazırlanan gerçek veri seti sisteme dahil edilmektedir. Bu çalışmada "Öğrencilerin Akademik Performans Veri Seti (xAPI-Edu-Verileri)" kullanılmaktadır [38]. Sonraki aşamada, öğrenci veri seti için ön işleme yapılmaktadır. Veri ön işleme, öznelik mühendisliği, model seçimi, hiper-parametre optimizasyonu ve lider algoritmanın seçilmesi adımları AutoML tarafında otomatik olarak, insan müdahalesi gerekmeden gerçekleştirilmektedir. Veriler %60 eğitim, %20 doğrulama ve %20 test verileri olarak ayrılmaktadır. AutoML sonucunda DRF algoritması lider algoritma olarak seçilmektedir. DRF algoritması, tek bir sınıflandırma veya regresyon ağacı yerine bir sınıflandırma veya regresyon ağaçları ormanı oluşturmaktadır. Bu ağaçların her biri, zayıf öğrencilerden (*weak learners*) oluşmaktadır. Hem sınıflandırma hem de regresyon yönteminde, nihai bir tahminde bulunmak için tüm ağaçların ortalama tahmini alınmaktadır [39]. H2O aracında AutoML yönteminin mevcut sürümünde DRF algoritması için ızgara araması yapılmamaktadır. Bu nedenle, bu çalışmada AutoML'den bağımsız olarak ayrıca

DRF algoritmasının hiper-parametrelerini optimize etmek amacıyla ızgara araması uygulanmaktadır. En son aşamada ise test veri seti kullanılarak önerilen algoritmanın performansı değerlendirilmektedir.



Şekil 3. Önerilen yöntemin akış diyagramı.

## 4. Bulgular ve Tartışma

### 4.1. Veri Seti

Bu çalışmada kullanılan "Öğrencilerin Akademik Performans Veri Seti (xAPI-Edu-Verileri)" Kalboard 360 E-öğrenme sisteminden toplanmaktadır [38]. Veri seti 17 öznitelik ve 480 öğrenci kaydından oluşmaktadır. Öznitelikler üç esas kategoriye ayrılmaktadır.

- Demografik öznitelikler: Cinsiyet, milliyet ve doğum yeri bilgilerinden oluşmaktadır.
- Akademik arka plan öznitelikleri: Öğrencilerin eğitim aşaması, sınıf düzeyi, bölüm ve dönem gibi arka plan özniteliklerini temsil etmektedir.
- Davranışsal öznitelikler: Sınıfta el kaldırma sayısı, kaynakları açma, ebeveynlerin anketlerini yanıtlama ve okul memnuniyeti gibi davranışları örneklemektedir.

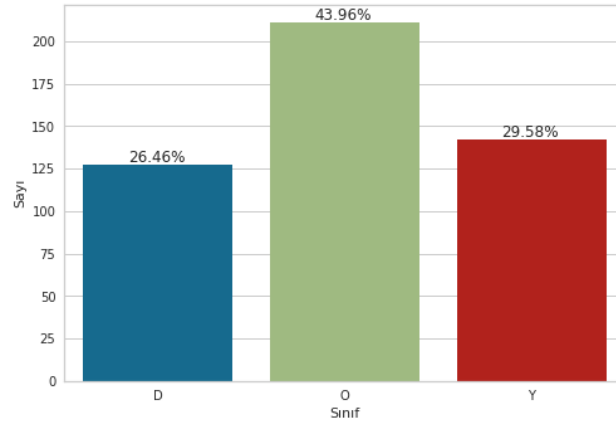
Tablo 3, bu çalışmada kullanılan veri setine ait özniteliklere genel bir bakış içermektedir. Veriler "Nominal" ve "Sayısal" olarak iki ana öznitelik türünden ibarettir. Verilerde eksik değer bulunmamaktadır.

Veri setinde tahmin edilecek hedef değişkeni olan 'Sınıf', kursun öğrenci notlarından oluşmaktadır. Düşük Seviye: 0 ile 69 arasındaki değerleri içermektedir, Orta Seviye: 70 ile 89 arasındaki değerleri içermektedir, Yüksek Seviye: 90-100 arasındaki değerleri içermektedir.

**Tablo 3. Veri Seti**

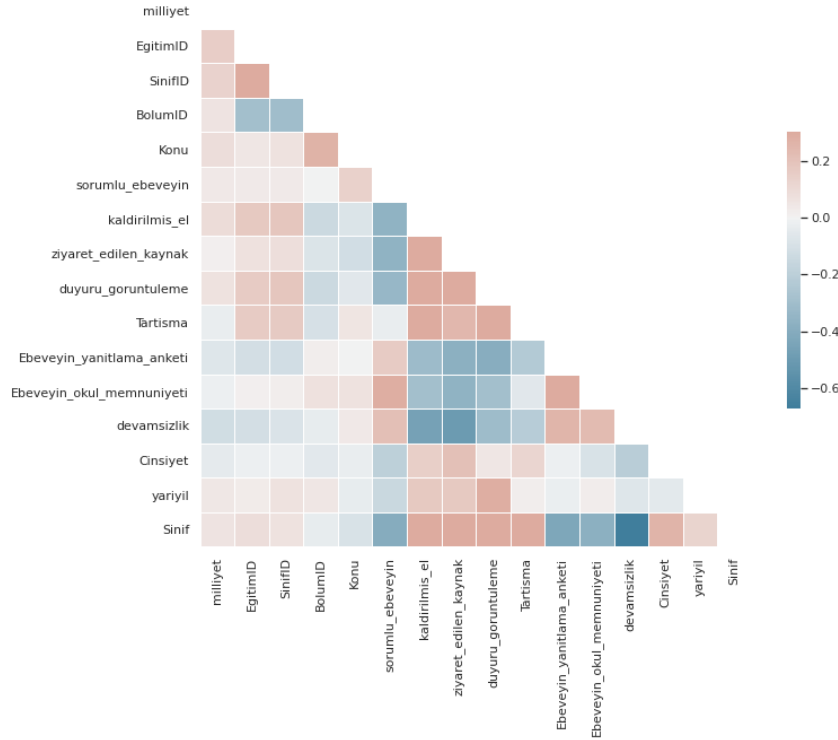
Öznitelik	Açıklama	Tür
Cinsiyet	Öğrencinin cinsiyeti	Nominal: 'Erkek' veya 'Kadın'
Milliyet	Öğrencinin uyruğu	Nominal: 'Kuveyt', 'Lübnan', 'Mısır', 'Suudi Arabistan', 'ABD', 'Ürdün', 'Venezuela', 'İran', 'Tunus', 'Fas', 'Suriye', 'Filistin', 'Irak', 'Libya'
Doğum yeri	Öğrencinin doğum yeri	Nominal: 'Kuveyt', 'Lübnan', 'Mısır', 'Suudi Arabistan', 'ABD', 'Ürdün', 'Venezuela', 'İran', 'Tunus', 'Fas', 'Suriye', 'Filistin', 'Irak', 'Libya'
Eğitim aşaması	Öğrencinin eğitim seviyesi	Nominal: 'Alt seviye', 'Ortaokul', 'Yüksekokul'
Sınıf seviyeleri	Öğrencinin sınıf seviyesi	Nominal: 'G-01', 'G-02', 'G-03', 'G-04', 'G-05', 'G-06', 'G-07', 'G-08', 'G-09', 'G-10', 'G-11', 'G-12'
Bölüm Kimliği	Öğrencinin sınıfı	Nominal: 'A', 'B', 'C'
Konu	Ders konusu	Nominal: 'İngilizce', 'İspanyolca', 'Fransızca', 'Arapça', 'BT', 'Matematik', 'Kimya', 'Biyoloji', 'Bilim', 'Tarih', 'Kuran', 'Jeoloji'
Yarıyıl	Okul yarıyıl dönemi	Nominal: 'Birinci', 'İkinci'
Sorumlu ebeveyn	Öğrenciden sorumlu ebeveyn	Nominal: 'Anne' veya 'Baba'
Kaldırılmış el	Öğrencinin sınıfta elini kaldırma sayısı	Sayısal: 0-100 arası
Ziyaret edilen kaynaklar	Öğrencinin bir kurs içeriğini ziyaret etme sayısı	Sayısal: 0-100 arası
Duyuruları görüntüleme	Öğrencinin yeni duyuruları kontrol etme sayısı	Sayısal: 0-100 arası
Tartışma grupları	Öğrencinin tartışma gruplarına katılma sayısı	Sayısal: 0-100 arası
Ebeveyn Yanıtlama Anketi	Ebeveyn okuldan sağlanan anketleri cevapladı	Nominal: 'Evet', 'Hayır'
Ebeveyn Okul Memnuniyeti	Ebeveynlerin okuldan memnuniyet derecesi	Nominal: 'Evet', 'Hayır'
Öğrenci Devamsızlık Günleri	Her öğrenci için devamsızlık gün sayısı	Nominal: 7'nin üstünde, 7'nin altında
Sınıf	Kursun öğrenci notu	Nominal: Düşük Seviye, Orta Seviye veya Yüksek Seviye

Şekil 4 ile veri seti üzerinde 'Sınıf' hedef değişkeninin dağılımı gösterilmektedir. Şekil 4'de sınıf bazında ağırlık oranlarında farklar görülmektedir.



**Şekil 4.** Veri seti üzerinde sınıf dağılımları

Şekil 4’de ‘D’ düşük seviye, ‘O’ orta seviye, ‘Y’ ise yüksek seviyeyi ifade etmektedir. Şekil 4 incelendiğinde veri setinde orta seviye öğrencilerin üstünlük sağladığı görülmektedir. Şekil 5’de ise öznitelikler arasındaki korelasyon gösterilmektedir.



**Şekil 5.** Öznitelikler arasındaki korelasyon

Şekil 5 incelendiğinde Sınıf değişkeni ile en yüksek pozitif korelasyonun Öğrenci Devamsızlık Günleri, Ziyaret edilen kaynaklar ve Öğrencinin derste el kaldırma sayısı değişkenleri arasında olduğu görülmektedir. Genel olarak Ziyaret edilen kaynaklar ve Öğrencinin derste el kaldırma sayısı arasında, Duyuruları görüntüleme ve Ziyaret edilen kaynaklar, Öğrencinin derste el kaldırma sayısı arasında pozitif korelasyon gözlemlenmektedir. Sınıf seviyeleri ve Bölüm kimliği öznitelikleri arasında ise negatif korelasyon görülmektedir. Tablo 4’te önerilen yöntem için seçilen en iyi öznitelikler yer almaktadır. Bu özniteliklerin seçimi Öznitelik Mühendisliği başlığı altında detaylandırılmıştır. Çalışmada önerilen DRF algoritması için en önemli öznitelikler, her bir değişkenin görel etkisi hesaplanarak belirlenmektedir.

**Tablo 4.** Önerilen yöntem için en önemli öznitelikler

Önem Sırası	Öznitelikler
1	Kaldırılmış el
2	Ziyaret edilen kaynak
3	Devamsızlık
4	Duyuru görüntüleme
5	Konu
6	Milliyet
7	Tartışma
8	Ebeveyn yanıtlama anketi
9	SınıfID
10	Sorumlu ebeveyn
11	Cinsiyet
12	Ebeveyn okul memnuniyeti
13	BölümID
14	Yarıyıl
15	EğitimID

## 4.2 Deney Sonuçları

Bu çalışmada, öğrencilerin akademik performanslarını tahmin etmek amacıyla AutoML yöntemi kullanarak kullanıcı müdahalesi gerektirmeden önışlem aşamaları gerçekleştirilmiş ve veri seti için en uygun model belirlenmiştir. Tüm deneyler, 8 GB RAM'e sahip (Dual-Core Intel Core i5, 1.8GHz) MacOS işletim sistemine sahip bilgisayarda Python'un 3.7.9 sürümü kullanılarak gerçekleştirilmektedir. AutoML ile veri önışleme, en uygun modelin seçimi otomatik olarak gerçekleştirilmektedir. Model performansını iyileştirmek için ızgara arama algoritması ile hiperparametre optimizasyonu yapılmaktadır. Tablo 5'de AutoML işleminin sonucunda lider algoritma dahil beş en iyi sonuç veren algoritmalar gösterilmektedir.

**Tablo 5.** AutoML sonucu

Model	MPCE	LOGLOSS	RMSE	MSE
DRF_1_AutoML	0.224	0.789	0.441	0.195
Derin Öğrenme_1_AutoML	0.249	0.629	0.439	0.193
Yığınlama topluluk_tüm modeller_AutoML	0.256	0.586	0.432	0.186
Derin öğrenme_grid_1_AutoML	0.257	1.590	0.472	0.223
GBM_1_AutoML	0.259	1.146	0.481	0.231

Tablo 5'de modeller, problem türüne göre varsayılan bir metrik temel alınarak sıralanmaktadır. İkili sınıflandırma (*binary classification*) problemlerinde, bu metrik AUC'dir. Çok-sınıflı sınıflandırma (*multi-class classification*) problemlerinde ise bu metrik sınıf başına ortalama hatadır (MPCE). Ayrıca, log kaybı (LOGLOSS), Kök Ortalama Kare Hata (RMSE) ve Ortalama Kare Hata (MSE) gibi bazı ek ölçümler de sağlanmaktadır [39]. AutoML yönteminde LOGLOSS değerleri otomatik olarak oluşturulmaktadır. LOGLOSS değerlerinin kaç iterasyon sonucu elde edildiği çıktı olarak verilmemektedir.

**Tablo 6.** DRF algoritmasının varsayılan hiper-parametre değerleri

Hiper-paramerte	Değer
number_of_trees (ağaç sayısı)	50
number_of_internal_trees (dahili ağaç sayısı)	150
min_depth (minimum derinlik)	6
max_depth (maksimum derinlik)	17
min_leaves (minimum yaprak)	15
max_leaves (maksimum yaprak)	72

Bütün bu işlemler AutoML içerisinde otomatik olarak gerçekleştirilmektedir. AutoML yönteminin ayarlanabilir parametrelerine *max\_model* ve *max\_runtime\_secs* parametreleri dahildir. *Max\_runtime\_secs*, AutoML yönteminin çalışacağı maksimum süreyi belirtmektedir. Bu parametre kullanıcı tarafından ayarlanmamışsa dinamik olarak 1 saate ayarlanmaktadır [39]. Bu çalışmada *max\_runtime\_secs* parametresi 300 sn. olarak belirlenmiştir. *Max\_models*, AutoML yönteminde oluşturulacak maksimum model sayısını belirtmektedir. Çalışmamızda bu değer 20 olarak ayarlanmaktadır. Tablo 5’de AutoML sonucunda DRF algoritmasının lider algoritma olarak seçildiği görülmektedir. Diğer iyi sonuç veren algoritmalar ise sırasıyla derin öğrenme, tüm modelleri içeren yığınlama topluluk modeli, ızgara araması ile hiper-parametreleri optimize edilmiş olan derin öğrenme ve GBM algoritmalarıdır. AutoML ile elde edilen lider algoritma için varsayılan hiper-parametreler ise Tablo 6’da verilmektedir.

AutoML sonucunda lider algoritma olan DRF için test veri seti için varsayılan hiper-parametreler ve ızgara aramasından sonra optimize edilen hiper-parametreleri ile elde edilen performans değerleri sırasıyla Tablo 7 (a) ve (b)’de verilmektedir.

**Tablo 7.** AutoML performans değerleri (a) DRF algoritmasının varsayılan hiper-parametreleri ile elde edilen sonuçlar (b) DRF algoritmasının ızgara aramasından sonra optimize edilen hiper-parametreleri ile elde edilen sonuçlar

	Doğruluk	Kesinlik	Özgüllük	F-skor
(a)	%77.50	%80.08	%80.70	%80.01
(b)	%82.30	%83.40	%82.40	%82.50

Tablo 7 (a) incelendiğinde AutoML ile elde edilen lider algoritma olan DRF algoritması ile doğruluk değerinin %77.5, kesinlik değerinin %80.08, özgüllük değerinin %80.7 ve f-skor değerinin ise %80.01 olduğu görülmektedir. DRF algoritması için AutoML yönteminin mevcut sürümünde ızgara araması yapılmamaktadır. Bu nedenle bu çalışmada, DRF algoritmasının hiper-parametrelerini optimize etmek amacıyla AutoML yönteminden bağımsız olarak ayrıca ızgara araması uygulanmaktadır. Tablo 7 (b)’de DRF algoritmasının ızgara araması ile elde edilen performans değerleri verilmektedir. Tablo 7 (b) incelendiğinde DRF algoritmasına ızgara araması uygulandıktan sonra performans değerlerinin arttığı gözlemlenmektedir. ızgara araması ile DRF algoritması için elde edilen optimal hiper-parametreler ise sırasıyla *max\_depth=6*, *n\_trees=100*, *sample\_rate=1* olarak hesaplanmaktadır.

Bu çalışmada önerilen algoritma, literatürde popüler olarak kullanılan KNN ve SVM algoritmaları ile kıyaslanmıştır [40]. AutoML yönteminde otomatik olarak yerine getirilen işlemler bu aşamada manuel olarak gerçekleştirilmektedir. İlk önce veri bağımlı ve bağımsız değişkenlere ayrılmaktadır.

Veri setinde bağımlı öznitelik olarak ‘Sınıf’ özneliği seçilmektedir. Veri setindeki kategorik değişkenler ‘LabelEncoder’ kullanılarak sayısal değişkenlere dönüştürülmektedir. Öğrencilerin performansını etkileyen öznelikleri seçmek amacıyla özyinelemeli özellik çıkarma (*Recursive feature extraction*, RFE) algoritması kullanılmaktadır. RFE yönteminde veri setindeki tüm öznelikler derecelendirilmekte ve en az skora sahip öznelikler, veri setinden çıkarılmaktadır. Tablo 8’de RFE yöntemi ile seçilen en önemli öznelikler gösterilmektedir. Tablo 4 ve Tablo 8 karşılaştırıldığında AutoML yöntemi ile hiçbir işlem gerektirmeden seçilen öznelikler ile RFE yöntemi kullanılarak seçilen özneliklerin aynı olduğu gözlemlenmektedir.

Bir sonraki adımda veriler eğitim ve test kümesine ayrılmaktadır. Verilerin %70’i eğitim, %30’u ise test için kullanılmaktadır. Öğrencilerin akademik performanslarını tahmin etmek amacıyla KNN ve SVM algoritmaları kullanılarak modeller oluşturulmaktadır. İlk olarak, KNN ve SVM algoritmalarının varsayılan hiper-parametreleri ile modeller oluşturulmaktadır. Daha sonra ızgara araması algoritması uygulayarak algoritmaların hiper-parametreleri optimize edilmektedir. Tablo 9’da hem de KNN hem de SVM algoritması için ızgara araması sonucu bulunan hiper-parametreler verilmektedir.

**Tablo 8.** RFE yöntemi ile seçilen en önemli öznelikler

Önem Sırası	Öznelikler
1	Cinsiyet
2	Milliyet
3	EğitimID
4	SınıfID
5	BölümID
6	Yarıyıl
7	Sorumlu ebeveyn
8	Ebeveyn yanıtlama anketi
9	Ebeveyn okul memnuniyeti
10	Devamsızlık
11	Kaldırılmış el
12	Ziyaret edilen kaynak
13	Konu
14	Duyuru görüntüleme
15	Tartışma

**Tablo 9.** Izgara araması ile KNN ve SVM algoritmaları için elde edilen optimal hiper-parametreler

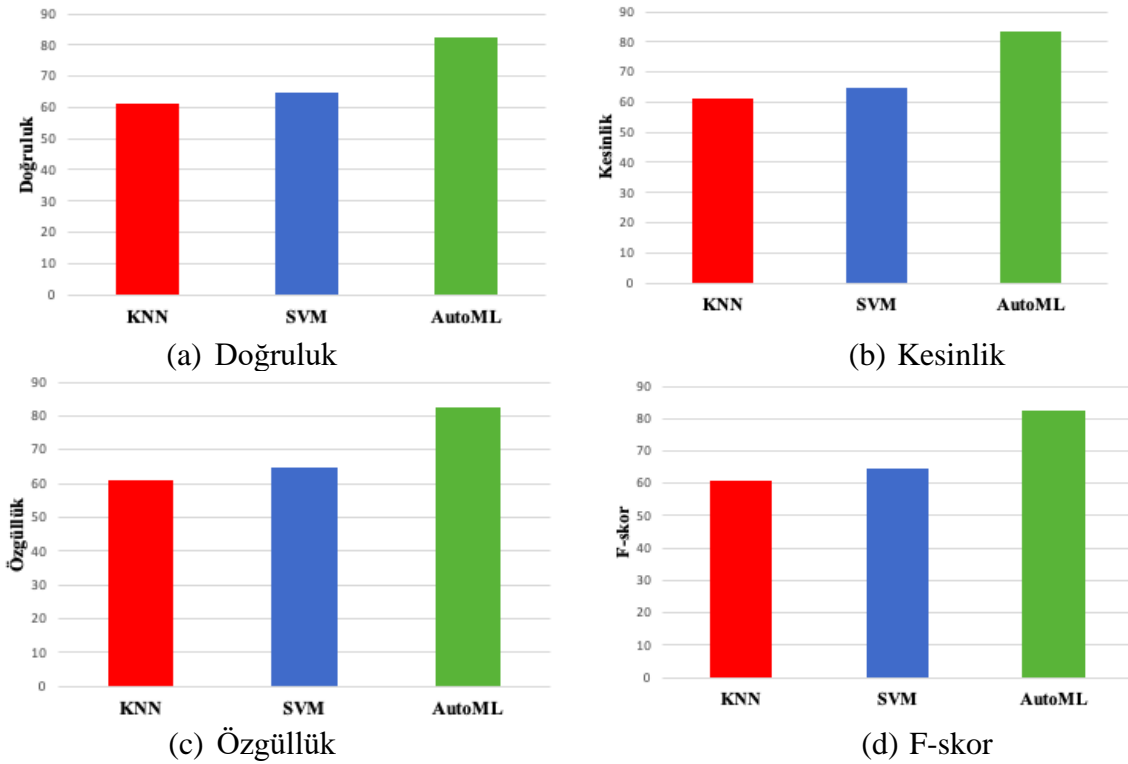
KNN	SVM
Metric: minkowski	C: 10
N_neighbors: 3	Gamma: 1
P: 1	Kernel: rbf

KNN ve SVM algoritmalarının varsayılan hiper-parametreler ve ızgara aramasından sonra optimize edilen hiper-parametreleri ile elde edilen performans değerleri sırasıyla Tablo 10 (a) ve (b)'de verilmektedir.

**Tablo 10.** KNN ve SVM algoritmalarının performans değerleri: (a) KNN ve SVM algoritmalarının varsayılan hiper-parametreleri ile elde edilen sonuçları, (b) KNN ve SVM algoritmalarının ızgara aramasından sonra optimize edilen hiper-parametreleri ile elde edilen sonuçları

		Doğruluk	Kesinlik	Özgüllük	F-skör
(a)	KNN	%53.40	%54.03	%53.40	%52.60
	SVM	%60.40	%60.90	%60.40	%59.90
(b)	KNN	%61.10	%61.05	%61.10	%61.01
	SVM	%64.50	%64.80	%64.50	%64.40

Tablo 10 (a) ve (b) incelendiğinde ızgara aramasından sonra her iki algoritmanın performans değerlerinin arttığı görülmektedir. Tablo 7 ile Tablo 10 karşılaştırıldığında ise AutoML yöntemi ile seçilen DRF algoritması hem varsayılan hem de optimize edilmiş hiper-parametreleri ile KNN ve SVM algoritmalarından daha yüksek sonuçlar elde edilmektedir. Şekil 6'da önerilen yöntemin performans değerleri ile KNN ve SVM algoritmasının performans değerleri karşılaştırılmaktadır.



**Şekil 6.** Önerilen yöntem ile KNN ve SVM algoritmalarının performans karşılaştırılması

Şekil 6 incelendiğinde önerilen yöntem ile elde edilen doğruluk, kesinlik, özgüllük ve f-skör değerleri geleneksel makine öğrenme algoritmalarından olan KNN ve SVM algoritmaları ile elde edilen değerlerden daha yüksek olduğu görülmektedir.

## 5. Sonuç ve Öneriler

Yapay zekâ destekli eğitim, akıllı eğitimi, yenilikçi sanal öğrenmeyi, tahmini ve veri analizini içermektedir. Eğitimde yapay zekanın ana senaryoları öğrencilerin ve okulların değerlendirilmesi, ödev ve sınavların notlandırılması ve değerlendirilmesi, kişiselleştirilmiş akıllı öğretim, akıllı okul,



çevrimiçi ve mobil eğitim olarak listelenebilir. Makine öğrenimi, veri madenciliği, bilgi modeline dayalı öğrenme için yapay zekâ sistemine çeşitli teknikler dahil edilmiştir. Makine öğrenimi, öğrenme analitiği ve veri madenciliği, eğitimle yakından ilişkili teknolojilerdir. Eğitsel ortamlardan gelen veriler ile anlamlı bilgi elde etmek için makine öğrenmesi algoritmaları kullanılmaktadır. Makine öğrenmesine dayalı çalışmalarda modelin üretim yolu farklı aşamalardan ve zorluklardan geçmektedir. AutoML yöntemi belirli görevlerde en iyi performansı elde etmek için veri ön işleme, model sınıfının seçimi, tahmin edicinin seçilmesi ve eğitimi, performansın değerlendirilmesi ve sonuçların yorumlanması işlemlerini otomatik olarak yerine getirmeyi amaçlamaktadır.

Bu çalışmada, öğrencilerin akademik performanslarını tahmin etmek amacıyla AutoML yönteminin etkinliği araştırılmaktadır. Kullanıcı müdahalesi olmadan veri ön işleme, öznelik seçimi, model seçimi ve modelin hiper-parametrelerinin ayarlanması bu yöntem ile otomatik olarak gerçekleştirilmektedir. Böylece bu alanda çalışan ve uzman olmayan araştırmacılar AutoML tekniğini kullanarak kendi veri setleri için en uygun modeli hızlıca belirleyebilirler. Deneysel sonuçlarında, AutoML yöntemi ile veri seti için en iyi performans değerlerini veren beş algoritma belirlenmektedir. Bu algoritmalar içerisinde, lider algoritma olarak DRF algoritması seçilmektedir. AutoML yönteminde, DRF algoritması için hiper-parametre optimizasyonu yapılmamaktadır. Bu nedenle, algoritmanın hiper-parametrelerini optimize etmek amacıyla ızgara araması uygulanmaktadır. Önerilen algoritma ile öğrencilerin akademik performansını tahmin etmede daha başarılı sonuçlar verdiği gözlenmektedir. AutoML ile daha az kullanıcı müdahalesi ile üzerinde çalışılan veri seti için ön işleme gerektirmeden lider algoritma ve en uygun hiper-parametre seti seçimi otomatik olarak gerçekleştirilebilmektedir. Bu çalışmada, öğrenci verileri üzerinde makine öğrenimi deneyleri gerçekleştirirken AutoML yöntemi kullanmanın etkili olduğu görülmektedir. AutoML kullanarak seçilen model öğrencilerin performansının erken tahmin edilmesinde iyi sonuçlar göstermektedir. Böylece, öğrencilere zamanında destek ve etkili müdahale stratejileri sağlanabilmektedir. Genel olarak, eğitim alanında AutoML yönteminin ve araçlarının hem uzmanlar hem de yeni araştırmacılar olmak üzere eğitim ortamlarından gelen verileri kullanarak akademik sonuçları iyileştirmek amacıyla yeni ufuklar açacağına inanıyoruz. AutoML ile etkileyici sonuçlar elde edilmiş olsa da iyileştirmeler yapılmakta ve yeni teknikler geliştirilmektedir. Gelecekte farklı AutoML teknikleri ve araçlarını kullanarak öğrencilerin akademik performans tahmin sonuçlarının iyileştirmesine yönelik çalışmalar yapılması planlanmaktadır.

### **Yazar(lar)ın Katkıları**

SA, bilimsel yayın araştırması, algoritmaların kodlanmasında ve test edilmesinde, makalenin yazımında; SBK, bilimsel yayın araştırması, problemin ve yöntemin tanımlanmasında ve oluşturulmasında, makalenin yazımında katkı sağlamışlardır.

Her iki yazar da makalenin son halini okudu ve onayladı.

### **Çıkar Çatışması**

Yazarlar, çıkar çatışması olmadığını beyan eder.

### **Kaynaklar**

- [1]. Ada, Ş., Başar, E., Dağlı, A., Ekinci, E., Ergün, M., Gelbal, S., Hoşgörür, V., Kiroğlu, K., Mahiroğlu, A., Taştan, N., Eğitim bilimine giriş, Pegem A Yayıncılık, Ankara. 2007.
- [2]. Romero, C., ve Ventura, S., Educational data mining and learning analytics: An updated survey, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2020, 10(3): e1355.

- [3]. Tomasevic, N., Gvozdencovic, N., Vranes, S., An overview and comparison of supervised data mining techniques for student exam performance prediction, *Computers & Education*, 2020, 143: 103676.
- [4]. Baradwaj, B.K., Pal, S., Mining educational data to analyze students' performance, *International Journal of Advanced Computer Science and Applications*, 2011, 6(2): 63-69.
- [5]. Salloum, S.A., Alshurideh, M., Elnagar, A., Shaalan, K., Mining in Educational Data: Review and Future Directions, *International Conference on Artificial Intelligence and Computer Vision*, 2020: 92-102.
- [6]. Pekuwali, A.A., Prediction of student learning outcomes using the Naive Bayesian Algorithm, (Case Study of Tama Jagakarsa University), In *IOP Conference Series: Materials Science and Engineering*, 2020, 823(1): 012056.
- [7]. Shrestha, S., Pokharel, M., Data Mining Applications Used in Education Sector, *Journal of Education and Research*, 2020, 10(2): 27-51.
- [8]. Liñán, L.C., Pérez, Á.A.J., Educational Data Mining and Learning Analytics: differences, similarities, and time evolution, *International Journal of Educational Technology in Higher Education*, 2015, 12(3): 98-112.
- [9]. Khan, A., Ghosh, S. K., Student performance analysis and prediction in classroom learning: A review of educational data mining studies, *Education and information technologies*, 2021, 26(1): 205-240.
- [10]. Ghorbani, R., Ghousi, R., Comparing different resampling methods in predicting students' performance using machine learning techniques, *IEEE Access*, 2020, 8(1): 67899-67911.
- [11]. Kaunang, F.J., Rotikan, R., Students' Academic Performance Prediction using Data Mining, *IEEE 2018 Third International Conference on Informatics and Computing (ICIC)*, 2018, 1-5.
- [12]. Asif, R., Hina, S., Haque, S.I., Predicting student academic performance using data mining methods, *International Journal of computer science and network security*, 2017, 17(5): 187-191.
- [13]. Quinn, R.J., Gray, G., Prediction of student academic performance using Moodle data from a Further Education setting, *Irish Journal of Technology Enhanced Learning*, 2020, 5(1): 17-29.
- [14]. Gandy, R., Kasper, D., Luna, A., Creating a Student Success Predictor using Statistical Learning, *Technical Report*, 2019.
- [15]. Ahammad, K., Chakraborty. B., Akter, E., Fomey, U.H., Rahman, S., A Comparative Study of Different Machine Learning Techniques to Predict the Result of an Individual Student using Previous Performances, *International Journal of Computer Science and Information Security (IJCSIS)*, 2021, 19(1): 5-10.
- [16]. Singh, B.C., Protikuzzaman, M.D., Baowaly, M.K., Devnath, M.K., Predicting Undergraduate Admission: A Case Study in Bangabandhu Sheikh Mujibur Rahman Science and Technology University, Bangladesh, *International Journal of Advanced Computer Science and Applications*, 2020. 11(12): 138-145.
- [17]. Wandera, H., Marivate, V., ve Sengeh, M.D., Predicting school performance using a combination of traditional and non-traditional education data from South Africa, *Technical Report*, 2019.
- [18]. Sagar, M., Gupta, A., Kaushal, R., Performance prediction and behavioral analysis of student programming ability, *IEEE 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2016: 1039-1045.
- [19]. Alshabandar, R., Hussain, A., Keight, R., Khan, W., Students Performance Prediction in Online Courses Using Machine Learning Algorithms, *IEEE 2020 International Joint Conference on Neural Networks (IJCNN)*, 2020: 1-7.
- [20]. Liu, W., Xu, W., Zhan, X., Liu, W., Cheng, W., Student Performance Prediction by LMS Data and Classroom Videos, *IEEE 2020 15th International Conference on Computer Science & Education (ICCSE)*, 2020: 535-539.

- [21]. Ha, D.T., Giap , C.N., Loan, P.H.T., Huong, N.T.L., An Empirical Study for Student Academic Performance Prediction Using Machine Learning Techniques, *International Journal of Computer Science and Information Security (IJCSIS)*, 2020, 18(3): 21-28.
- [22]. Asril, T., Isa, S.M., Prediction of Students Study Period using K-Nearest Neighbor Algorithm, *International Journal of Emerging Trends in Engineering Research*, 2020, 8(6): 2585-2593.
- [23]. Bunkar, K., Tanwani S., Student Performance Prediction Using C4.5 Decision Tree and CART Algorithm, 2020, 2(9): 1702-1716.
- [24]. Farissi, A., ve Dahlan, H.M., Genetic Algorithm Based Feature Selection With Ensemble Methods for Student Academic Performance Prediction, *Journal of Physics: Conference Series*, 2020, 1501(1): 110-117.
- [25]. Abbasoğlu, B., Ortaokul Öğrencilerinin Akademik Başarılarının Eğitsel Veri Madenciliği Yöntemleri ile Tahmini, *Veri Bilimi*, 2020, 3(1): 1-10.
- [26]. Sathe, M.T., Adamuthe, A.C., Comparative Study of Supervised Algorithms for Prediction of Students' Performance, *International Journal of Modern Education & Computer Science*, 2021, 13(1): 1-21.
- [27]. Tsiakmaki, M., Kostopoulos, G., Kotsiantis, S., Ragos, O., Implementing AutoML in educational data mining for prediction tasks, *Applied Sciences*, 2020, 10(1): 90-116.
- [28]. Drăgulescu, B., ve Bucos, M., Hyperparameter tuning using automated methods to improve models for predicting student success, *International Conference on Information and Software Technologies*, 2020: 309-320.
- [29]. Tsiakmaki, M., Kostopoulos, G., Kotsiantis, S., Ragos, O., Fuzzy-based active learning for predicting student academic performance using autoML: a step-wise approach, *Journal of Computing in Higher Education*, 2021: 1-33.
- [30]. Halvari, T., Nurminen, J.K., Mikkonen, T., Testing the Robustness of AutoML Systems, *International Conference on Tools with Artificial Intelligence*, 2020: 103-116.
- [31]. He, X., Zhao, K., Chu, X., AutoML: A Survey of the State-of-the-Art', *Knowledge-Based Systems*, 2021, 212: 106622.
- [32]. LeDell, E., Poirier, S., H2O automl: Scalable automatic machine learning, in *Proceedings of the AutoML Workshop at ICML*, 2020.
- [33]. Truong, A., Walters, A., Goodsitt, J., Hines, K., Bruss, C.B., Farivar, R., Towards automated machine learning: Evaluation and comparison of AutoML approaches and tools, *IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, 2019: 1471-1479.
- [34]. Zöllner, M.A., ve Huber, M.F., Benchmark and survey of automated machine learning frameworks, *Journal of Artificial Intelligence Research*, 2021, 70(1): 409-472
- [35]. Yao, Q., Wang, M., Chen, Y., Dai, W., Li, Y.F., Tu, W.W., Yang, Q., Yu, Y., Taking human out of learning applications: A survey on automated machine learning, 2018, arXiv preprint arXiv:1810.13306.
- [36]. Visa, S., Inoue, A., Ralescu, A., Confusion Matrix-based Feature Selection, *MAICS*, 2011, 710 (1): 120-127.
- [37]. Cortes, C., ve Mohri, M., 'AUC optimization vs. error rate minimization', *Advances in neural information processing systems*, 2003, 16(1): 313-320.
- [38]. Moisa, V., 'Adaptive learning management system', *Journal of Mobile, Embedded and Distributed Systems*, 2013, 5(2): 70-77.
- [39]. <https://docs.h2o.ai/>, Erişim tarihi 23.04.2021.
- [40] Uğuz, S., Makine öğrenmesi teorik yönleri ve Python uygulamaları ile bir yapay zekâ ekolü, Nobel Yayıncılık, Ankara. 2019.