

PISA 2015 Veri Setinde OVA ve OVO Stratejileri Çerçevesinde Bazı Temel Sınıflandırıcıların Performanslarının Karşılaştırılması

Hümeýra DEMİR¹  ve Gürol ZIRHLIOĞLU² 

Öz: Bu çalışmanın amacı hem çok sınıflı verilerin sınıflandırılması için kullanılan OVA ve OVO stratejilerinin hem de bu stratejiler altında uygulanan bazı ikili sınıflandırıcıların performanslarının karşılaştırılmasıdır. Çalışmanın yöntemi betimsel araştırmadır. 5895 örnek içeren veri setinden, boş veri içeren örnekler silindiğinde elde edilen 26 bağımsız 1 bağımlı değişkenden oluşan 3459 örneklı veri setine OVA ve OVO stratejileri altında belirlenen beş farklı algoritma (J-48, KNN, LR, NB, DVM) uygulanmıştır. Test seçeneklerinden 10 katlı çapraz geçirme, birleştirme stratejilerinden de oy verme tekniği kullanılmıştır. Analiz sonucunda OVA ve OVO stratejilerinin her ikisi altında en başarılı algoritmalar LR ve NB algoritmaları iken en başarısız algoritma KNN algoritması olarak belirlenmiştir. En az örnek içeren sınıfı tahminleyebilme başarısı bakımından her iki strateji altında da en başarılı algoritma KNN algoritmasıdır. Algoritmalar doğruluk değeri ve hata ortalaması bakımından OVA stratejisi altında F metriği bakımından ise OVO stratejisi altında daha başarılıdır. Farklı algoritmaların performansları farklı koşullarda denenebilir. Eğitim ortamlarından gelen verilerin bulunduğu veritabanlarındaki çok kategorili verilere OVA ve OVO stratejileri ile çalışmada kullanılan algoritmalar uygulanarak alınacak kararların daha isabetli olması sağlanabilir.

Anahtar kelimeler: Ayırıştırma stratejileri, çok sınıflı sınıflandırma, sınıflandırma algoritmaları, performans karşılaştırma

Comparing the Performance of Some Basic Classifiers Within the Framework of OVA and OVO Strategies in PISA Datasets

Abstract: This paper compares the OVA and OVO classification strategies used for multi-class data classification, as well as the effectiveness of specific binary classifiers using these procedures. The study utilizes descriptive research as its methodology. Science achievement test data and survey results of 5895 Turkish students who participated in PISA-2015 were used. Five different algorithms (J-48, KNN, LR, NB, DVM) determined under the OVA and OVO strategies were applied to the data set with 3459 samples, consisting of 26 independent and 1 dependent variable, obtained when the samples with null data were deleted from the data set containing 5895 samples. 10-fold cross-validation from the test options and voting techniques from the joining strategies were used. The LR and NB algorithms are the most effective under

Geliş tarihi/Received: 11.02.2022

Kabul Tarihi/Accepted: 01.09.2022

Makale Türü: Araştırma Makalesi

*Bu çalışma ikinci yazar danışmanlığında birinci yazarın “PISA 2015 Verisetinde OVA ve OVO Stratejileri Çerçevesinde Bazı Temel Sınıflandırıcıların Performanslarının Karşılaştırılması ” adlı yüksek lisans tezinin bir kısmından üretilmiştir.

¹ Arş. Gör., Hacettepe Üniversitesi, Eğitim Bilimleri, hmeýrademir@gmail.com, 0000-0002-6300-2674

² Dr. Öğr. Üyesi, Van Yüzüncü Yıl Üniversitesi, Eğitim Bilimleri, gurol@yyu.edu.tr, 0000-0001-9949-0668

Atf için/To cite: Demir, H., & Zirhlioğlu, G. (2022). PISA 2015 veri setinde OVA ve OVO stratejileri çerçevesinde bazı temel sınıflandırıcıların performanslarının karşılaştırılması. *Van Yüzüncü Yıl Üniversitesi Eğitim Fakültesi Dergisi*, 19(3), 733-759. <https://doi.org/10.33711/yyuefd.1071860>

both OVA and OVO methods, whereas the KNN algorithm is the least effective. The KNN algorithm, which estimates the class with the fewest samples, performs best when both strategies are used. Algorithms that use the OVA strategy and the F metric for the OVO approach perform better in terms of accuracy and mean error metrics. The effectiveness of an algorithm can be evaluated using different scenarios. The study's OVA and OVO techniques and algorithms can be applied to multi-categorical data in databases that incorporate data from educational environments to help decision-makers make more accurate choices.

Keywords: Multiclass classification, binarization techniques, classification algorithms, comparing performance

Giriş

Veri tabanı ve bilgisayar teknolojilerindeki gelişmelerle beraber günlük yaşantımızda yaptığımız birçok işlem kayıt altına alınmaya başlanmış, kayıt altına alınan veriler büyük veri yığınları oluşturmuştur. Bu büyük miktardaki veriden anlamlı, işe yarar bilgiler ortaya çıkarma ve geleceğe yönelik tahminler yapabilme isteği veri madenciliği kavramını ortaya çıkarmıştır (Han vd., 2011). Veri madenciliği; büyük miktardaki veri içerisinde gizli kalmış, işe yarar, güvenilir, potansiyel olarak kullanışlı ve daha önceden bilinmeyen bilgilerin, örüntülerin, kuralların, bağıntıların çeşitli algoritmalar, istatistiksel ve matematiksel teknikler, desen tanımlayıcı teknolojiler ve bilgisayar programları kullanılarak çıkarılması işlemidir. (Akpınar, 2000; Babadağ, 2006; Fayyad vd., 1996; Hung vd., 2006). Her disiplinde toplanan büyük miktardaki veri, veri madenciliğinin birçok alanda uygulanmasını sağlamıştır. Bu disiplinlerden biri de eğitimidir.

Eğitsel veri madenciliği, eğitim ortamlarından gelen büyük boyutlardaki veriden bilgi elde etmek amacıyla yöntemler geliştiren ve bu yöntemler sayesinde öğrencileri ve öğrenme ortamlarını daha iyi anlamaya çalışan bir disiplindir (International Educational Data Mining Society [IEDMS], 2016). Günümüzde öğrencilere ait kişisel bilgiler, başarı durumları ve sınav notları gibi birçok veri kayıt altına alınmaktadır. Kayıt altına alınan bu verilerden elde edilecek bilgiler, eğitimdeki problemlerin tespitinde, eğitim kalitesinin artırılmasında kullanılabilir. (Özbay, 2015). Veri madenciliğinin eğitimde kullanılma sürecinde veriler sınıf ortamlarından toplanabileceği gibi uzaktan eğitim ortamlarından da toplanabilir (Zaiane & Luo, 2001). Eğitim verileriyle gerçekleştirilecek veri madenciliği çalışmaları ile, öğrencilerin başarılarına etki eden etmenler belirlenebilmekte, öğrenci profilleri belirlenerek gruplama yapılabilmekte, öğrenci başarı durumları ya da mezuniyet notları tahmin edilebilmekte ve kullanılacak birçok başka uygulama ile eğitimin kalitesi artırılabilir (Özbay, 2015).

Makine Öğrenmesi, verilen bir problemi probleme ait ortamdan edinilen veriye göre modelleyen çeşitli algoritmaları içeren bir disiplindir. Makine öğrenmesi ile bilgisayarların daha önceki örnekler yoluyla öğrenmesi sağlanmaktadır. Makine öğrenmesi tekniklerinin büyük veri tabanlarına uygulanması ise veri madenciliğidir. Makine öğrenmesi algoritmaları veri madenciliği sürecinin uygulama aşamasında yer almaktadır. Eğitim ortamından gelen bir veri seti üzerinde makine öğrenmesi algoritmaları aynı alandan yeni verilerin sınıfını tahmin edebilmek amacıyla kullanılabilir. Bağımlı değişkeni iki sınıf içeren veri setlerindeki sınıflandırma problemi, ikili sınıflandırma problemi; ikiden fazla sınıf içeren veri setlerindeki sınıflandırma ise çok sınıflı sınıflandırma problemi olarak adlandırılmaktadır (Lorena vd., 2008). İkili sınıflandırma tekniklerini kullanarak çok sınıflı problemleri ele almak için benimsenen iki yaklaşım bulunmaktadır. Bunlardan biri kullanılan algoritmanın iç operasyonlarının uyarlanması diğeri çok sınıflı problemin iki sınıflı sınıflandırma problemlerine ayrıştırılmasıdır. Herhangi bir ikili sınıflandırma algoritmasının çok sınıflı bir problem için genişletilmesi, bir diğer deyişle iç

operasyonlarının uyarlanması bazı durumlarda pratik olmamakta veya gerçekleştirmek kolay olmamaktadır (Passerini vd., 2004). Bu nedenle, çok sınıflı problemleri iki sınıflı alt problemlere ayırıştırmak yani ayırışma stratejilerini kullanmak daha yaygındır. Problem birkaç iki sınıflı alt probleme ayrıldıktan sonra her bir alt problem için kullanılan sınıflandırıcılardan elde edilen sonuçlar yapılacak çok sınıflı tahmin için birleştirilir. Ayırıştırmaya stratejilerinin çok sınıflı verilerin çözümlenmesinde kullanılmasının birkaç sebebi vardır. Ayırıştırmaya yaklaşımı sınıfları belirlemede karmaşıklığı azaltabilir ve aynı zamanda çok sınıflı problemler için genişletilebilen algoritmaların da kullanılabilmesini sağlar. Tüm sınıfları aynı anda ayıran bir sınıflandırıcı kullanmaktansa oluşturulan sınıf çiftleri için kullanılan doğrusal sınıflandırıcılardan elde edilen sonuçları birleştirmek çok daha basit bir alternatiftir (Lorena vd., 2008).

Çok sınıflı problemlerin ikili sınıflandırma problemlerine ayrılması için çeşitli alternatifler kullanılabilir. Genellikle bu ayırışmalar, Allwein vd. (2000) tarafından önerilen bir kod-matris çerçevesi kullanılarak yapılır. Burada kod-matris M ile gösterilir. Bu matrisin satırları her sınıf için atanan kodları içerir. M matrisinin sütunları ise iki sınıflı parçaları tanımlar ve ikili sınıflandırıcıların bu sınıflar için belirlediği etiketlerle temsil edilir. k , çoklu sınıf problemindeki sınıfların sayısını ve l , çok sınıflı problem çözümünde kullanılan ikili sınıflandırıcıların sayısını temsil etmek üzere M matrisi $k \times l$ boyutuna sahiptir.

Literatürde ayırıştırmaya stratejileri arasında en yaygın kullanılanlar OVA ve OVO stratejileridir. OVA stratejisinde k sınıflı bir problem verildiğinde k tane ikili sınıflandırıcı $f_i(x)$ üretilir. Bu tahminleyicilerden her biri i . sınıfı kalan sınıflardan ayırmak için eğitilir. OVO stratejisinde k sınıf için $k \cdot \binom{k-1}{2}$ tane ikili sınıf oluşturulur ve bu sayıda ikili sınıflandırıcı kullanılır. Bu sınıflandırıcıların her biri i ve j sınıf çiftlerini ayırmak için kullanılır (Arruti vd., 2014).

Literatürde son sınıfı belirlemek için önerilen bir dizi toplama (birleştirme) stratejisi vardır (Galar vd., 2015; Liu vd., 2008). Oylama stratejisi (Vote) basit ancak güçlü bir stratejidir. İkili oylama veya maksimum kazançlar kuralı olarak da adlandırılan oylama stratejisinde, tahmin edilen sınıfa ikili sınıflandırıcı tarafından bir oy verilir. Her sınıfın aldığı oylar sayılır ve en fazla oyu alan final sınıfı tahmin edilmiş olur (Friedman, 1996).

Çalışmada kullanılan sınıflandırıcılar J-48, k -en yakın komşuluk (KNN), Lojistik Regresyon (LR), Naive Bayes (NB) ve Destek Vektör Makineleri (DVM) algoritmalarıdır. J-48 algoritması veri madenciliğindeki sınıflandırma modeli içerisinde yer alan karar ağacı algoritmalarındandır. Sınıflama algoritmaları önceden sınıfları belli olan verileri kullanarak bir model oluşturur ve sınıfı bilinmeyen bir örneği doğru kategoriye yerleştirmeye çalışır. Karar ağaçları düğüm, yaprak ve dallardan meydana gelmektedir (Han vd., 2011). Veri setindeki her bir değişken bir düğümle, yapılan testler dallarla ve veri setindeki sınıflar yapraklarla temsil edilmektedir. Karar ağaçlarında her düğümde yapılan testler, verilere ilişkin bir dizi sorular sorulmasını ve cevaplar elde edilmesini içerir. Elde edilen cevaplar toplanarak karar kuralları oluşturulur. Ağacın ilk düğümü olan kök düğümünde sorular sorulmaya başlanır ve dalları olmayan düğümler bir diğer deyişle yapraklar elde edilene kadar devam eder (Pal & Mather, 2003). Karar ağaçları oluşturulurken ağaçtaki dallanmaların hangi kritere göre yapılacağı bir diğer deyişle düğüm noktalarındaki değişkenlerin sırasının belirlenmesi için geliştirilmiş çeşitli yaklaşımlar bulunmaktadır. Bilgi kazancı ve bilgi kazanç oranı, Gini indeksi, Twoing gibi yaklaşımlar bunların başlıcalarıdır. Karar ağacı algoritmaları ağaç yapısını oluştururken, bazı düğümlerde ortaya çıkan kategoriler çok az örnek içerdiğinde ağacın yapısı karmaşıklaşır bu sebeple sınıflandırma

doğruluğunu çok fazla etkilemeyen dallanmaların, ağaçtan çıkarılarak budama işlemi gerçekleştirilir. (Özkan, 2016).

KNN algoritması eski fakat popüler, iyi bilinen ve etkili sınıflandırma algoritmalarındadır (Batista & Silva, 2009; Bhatia & Vandana, 2010; Qiu vd., 2008). Bu algoritma sınıflandırma yaparken ele alınan örneğin k en yakın komşuluğundaki örnekleri dikkate alarak sınıflandırma yapar (Cover & Hart, 1967). Bir örneğin sınıfı belirlenmek istendiğinde bu örneğe en az uzaklığa sahip olan k tane örneğin çoğunlukla seçtiği sınıf bu örneğin sınıfı olarak belirlenir (Han vd., 2011). KNN algoritmasında sınıflandırılacak olan örneğin k yakınlığındaki komşularına olan uzaklığının nasıl ölçüleceği önemli bir parametredir. Bu uzaklık öklit uzaklığı ile ölçülebileceği gibi başka ölçüler de kullanılmaktadır (Coomans & Massart, 1982).

LR yöntemi neden-sonuç ilişkilerinin araştırıldığı bilimsel araştırmalarda kullanılmaktadır. Yordayıcı ve yordanan değişkenlerin yapısına göre kullanılan regresyon modeli farklılık göstermektedir (Kayri & Çokluk, 2010). Doğrusal regresyon modellerinde yordayıcı ve yordanan değişkenlerin sürekli değişken olması ve bazı sayıtların karşılanması gerekirken; LR, bağımlı değişkenin kategorik olduğu durumlarda kullanılmakta (Mertler & Vannata, 2005) ve doğrusal regresyondaki sayıtları gerektirmemektedir (Kılıç, 2000). Lojistik regresyonda örneklerin hangi gruba girdiğinin belirlenmesi amacıyla bir regresyon denklemi oluşturulmakta; bu sayede hem sınıflandırma yapılmakta hem de bağımlı ve bağımsız değişkenler arasındaki ilişkiler ortaya konulmaktadır (Mertler & Vannata, 2005). LR, katsayıların kestirilmesi amacıyla en çok olabilirlik yöntemini kullanmaktadır. Bu yaklaşımda amaç bir olayın gerçekleşme olasılığını maksimum yapmaktır (Hair vd., 2006). Elde edilen model için lojistik regresyonda olasılık, odds oranı ve odds oranının doğal logaritmasının (logit) hesaplanması gerekmektedir. Odds oranı bir olayın olma olasılığının olmama olasılığına oranıdır. Odds oranının logaritması ise lojit değeridir (Field, 2005).

NB sınıflama algoritması; Bayes teorisine dayanan, hem tahmin edici hem de tanımlayıcı işlemlerde kullanılmaktadır. Bu algoritma tüm değişkenlerin birbirinden bağımsız ve hepsinin aynı öneme sahip olduğu varsayımına dayanır (Han vd., 2011; Özkan, 2016; Zhang, 2004). NB algoritması, belirli bir sınıftaki bir özellik değerinin etkisinin, diğer özniteliklerin değerlerinden bağımsız olduğunu varsayar. Bu varsayım sınıf koşullu bağımsızlık olarak adlandırılır (Han vd., 2011). Bağımsızlık varsayımı NB sınıflandırıcısı için bazı avantajlar ve dezavantajlar ortaya çıkarmaktadır. Bağımsızlık varsayımı durumu çoğu zaman gerçek dışı olsa da NB algoritmasının sınıflamadaki başarısı ve diğer bazı sınıflama algoritmalarından üstünlüğü çeşitli çalışmalarla ortaya konulmuştur. Bağımsızlık varsayımı her bir değişkenin tek tek öğrenilmesine olanak vermektedir. Böylece, çok değişkene sahip olan verilerde bile sınıflama işleminin hızlı olmasına olanak sağlamaktadır (Aydoğan, 2008; Zhang, 2004).

Destek vektör makineleri (DVM) algoritması orijinal eğitim verilerini daha yüksek bir boyuta dönüştürmek için doğrusal olmayan bir dönüşüm kullanan ve bu yeni boyutta en uygun ayırıcı hiperdüzlemi arayan oldukça basit ve etkili bir sınıflandırma algoritmasıdır. Hiperdüzlem, bir sınıfı diğerinden ayıran bir karar sınırını ifade eder. Yeterince yüksek bir boyutta uygun doğrusal olmayan bir dönüşüm ile iki sınıftan alınan veriler her zaman bir hiperdüzlem ile ayrılabilir. DVM algoritması bu hiperdüzlemi bulmak için destek vektörlerini ve bu destek vektörleri tarafından tanımlanan marjinleri kullanır (Han vd., 2011). A_1 ve A_2 ile temsil edilen iki değişkene sahip bir veri setinin bulunduğu ve bu veri setindeki örneklerin doğrusal olarak (bir doğru yardımıyla) ayrılabilirdiği varsayıldığında sınıfları birbirinden ayırabilmek için çizilebilecek sonsuz sayıda çizgi vardır. Çizilebilecek en optimal ya da en uygun çizginin bulunması yani

oluşturulan modelin test verilerinde minimum hata oranına sahip olması beklenmektedir. Ele aldığımız veri seti üç boyutlu olursa yani üç değişken bulunursa bu durumda sınıfları en iyi ayıran düzlemin bulunması gerekir. N boyutlu bir veri seti için ise sınıfları en iyi ayıran hiperdüzlem bulunmalıdır. Hiperdüzlem terimi girdi değişkenlerin sayısına bakılmaksızın aranan karar sınırının belirlenmesi için kullanılmaktadır (Han vd., 2011).

Kullanılan algoritmaların performansları belirlenirken ve değerlendirilirken bazı metriklerden yararlanır. Algoritmaların performans değerlendirmesi için bu çalışmada doğruluk, kesinlik, duyarlılık, F istatistiği ve kappa istatistiği metrikleri ile modellerin hata miktarlarını değerlendiren metriklerden Ortalama Mutlak Hata (MAE), Görelî Mutlak Hata (RAE), Kök Ortalama Karesel Hata (RMSE), Ortalama Mutlak Görelî Hata (RRSE) kullanılmıştır. Bu değerlerin hesaplanabilmesi amacıyla karışıklık matrisi oluşturulur. Karışıklık matrisi hem algoritmanın öngörüsünü hem de gerçek sınıf bilgisini içermektedir (Coşkun & Baykal, 2011).

Model başarımının ölçülmesinde kullanılan en popüler ve basit yöntem, modele ait doğruluk oranıdır. Doğru sınıflandırılmış örnek sayısının toplam örnek sayısına oranı olarak tanımlanmıştır. Hata oranı ise bu değerın 1'e tamlayanıdır. Diğer bir ifadeyle yanlış sınıflandırılmış örnek sayısının, toplam örnek sayısına oranıdır. Kesinlik, sınıfı 1 olarak tahminlenmiş gerçek pozitif örnek sayısının, sınıfı 1 olarak tahminlenmiş tüm örnek sayısına oranıdır. Duyarlılık, doğru sınıflandırılmış pozitif örnek sayısının toplam pozitif örnek sayısına oranıdır. F-Ölçütü, kesinlik ve duyarlılık metriklerinin harmonik ortalamasıdır (Coşkun & Baykal, 2011). Kappa istatistiği ise veri madenciliğinde sınıflandırma algoritmasının doğruluğu ile tesadüfi tahminde bulunan rastgele bir sınıflandırma algoritmasının doğruluğunun karşılaştırılması amacıyla kullanılmaktadır. Kappa istatistiği değerinin 1'e yaklaşması doğruluk değerinin bağımlı değişkenin sınıflarındaki örnek sayısındaki yığılmaya bağlı olmadığını yani ortaya çıkan başarımın rastgele olmadığını göstermektedir (Kalıpsız & Cihan, 2015).

Hata bir örneğin tahmin edilen değeri ile gerçek değeri arasındaki fark olarak tanımlanmaktadır. Hataların bir kayıp fonksiyonu ile temsil edildiği düşünülürse tüm örnekler üzerinden elde edilen hataların ortalaması ortalama mutlak hata, elde edilen hataların kareleri ortalaması ise ortalama karesel hata olarak tanımlanır. Ortalama karesel hatanın karekökü alındığında ortaya çıkan hata ölçüsü ortalama karesel hatanın karekökü (RMSE)'dir. Bu hata türü, ölçülen hatanın, tahmin edilen miktarla aynı büyüklükte olmasına izin vermesi bakımından önemlidir. Hata \bar{y} (eğitim verilerindeki y_i değerlerinin ortalaması) cinsinden görelî olarak belirlenmek istendiğinde kullanılan bazı başka hata ölçüleri de bulunmaktadır. Toplam kayıp, her zaman ortalamayı tahmin etmekten kaynaklanan toplam zarara bölünerek normalize edilebilir. Bu görelî hata ölçüleri rölâtif mutlak hata (RAE) ve görelî karesel hata (RSE) olarak adlandırılır. Rölâtif karesel hatanın karekökü (RRSE) değerinin elde edilmesi için RSE değerinin karekökü alınır. Böylece ortaya çıkan hata ile öngörülen miktar aynı büyüklükte olur (Han vd., 2011).

Lorena vd. (2008) çalışmalarında ikili sınıflandırıcıların çok sınıflı problemlerin çözümü için kullanılması üzerinde durmuşlar ve ayrıştırma teknikleri kullanılmadığında ikili sınıflandırıcıların çok sınıflı veriler için tekrar formüle edilmesi gerektiği sonucuna varmışlardır. Taruna ve Pandey (2014) çalışmalarında öğrencilerin notlarını tahminlemek amacıyla dört sınıflı bir veri seti üzerinde beş algoritmanın performanslarını kıyaslamışlar ve en başarılı algoritmaların KNN, karar ağacı ve Bayes Net algoritmaları olduğunu belirtmişlerdir. Sen vd. (2016) ayrıştırma tekniğini boosting ve aşırı örnekleme metotlarıyla birleştirmişler ve yeni bir teknik önermişlerdir. Önerdikleri tekniği OVA stratejisine dayalı olarak geliştirmişlerdir. Geliştirilen yöntem OVA

stratejisinde ortaya çıkan dengesiz sınıf dağılımı problemini çözmek için aşırı örnekleme kullanılmaktadır. Bu yeni tekniği birçok algoritma ile test etmişler ve muadilleriyle karşılaştırdıklarında geliştirdikleri yöntemin daha iyi performans gösterdiğini kanıtlanmışlardır. Galar vd. (2011) yaptıkları çalışmada C4.5, KNN, Ripper gibi bazı çok bilinen sınıflandırma algoritmalarını kullanarak OVO ve OVA stratejileri çerçevesinde farklı toplulaştırma (aggregation) tekniklerini kıyaslamışlardır. Sonuç olarak DVM ve PFDC algoritmalarının OVO stratejisi kullanıldığında daha iyi çalıştığını aynı şekilde C4.5 ve Ripper algoritmalarının da OVO şeması altında orijinal sınıflandırıcıya göre daha yüksek performans gösterdiğini, KNN algoritması kullanılmasının hem OVO hem OVA stratejilerinde orijinal algoritmanın performansını artırdığını fakat bunun istatistiksel olarak anlamlı olmadığını tespit etmişlerdir. Polat ve Güneş (2007) çalışmalarında çok sınıflı problemler için OVA stratejisini ve C4.5 karar ağacı yöntemini birleştirerek sınıflandırma başarısını büyük oranda artırmışlardır. Adnan ve Islam (2015) çalışmalarında bir karar ormanı oluşturma stratejisi olan rastgele ormanlar algoritmasını OVA şemasında temel algoritma olarak kullanarak farklı veri setlerinde kıyaslama yapmışlardır ve rastgele ormanlar algoritmasının OVA şemasında etkili bir yöntem olduğunu belirtmişlerdir. Varpa vd. (2011) çalışmalarında çok sınıflı bir veriyi OVO ve OVA stratejileri kullanarak iki sınıflı alt problemlere bölmüş daha sonra KNN ve DVM algoritmalarını bu alt problemlere uygulamışlardır. Ayırıştırma stratejileri kullanıldığında yeniden formüle etme yöntemine kıyasla, algoritmaların performanslarının daha iyi olduğunu ve genel olarak OVO stratejisinin OVA stratejisinden daha iyi çalıştığını ortaya koymuşlardır. En yüksek sınıflandırma doğruluğunun ise OVO stratejisi ile birlikte kullanılan KNN algoritmasıyla sağlandığını ifade etmişlerdir. Sharma ve Sahni (2011) çalışmalarında ID3, J48, basit sınıflandırma ve regresyon ağacı (simple CART) ve alternatif karar ağacı olmak üzere dört algoritmayı WEKA ortamında sınıflandırma doğruluğu açısından karşılaştırmışlardır. Simülasyon sonuçlarına göre J48 sınıflandırıcısı %92.7624 doğru sınıflandırma oranıyla, ID3, CART ve alternatif karar ağacını (ADTree) geride bırakmaktadır.

Çalışmanın Amacı

Bu çalışmanın amacı hem çok sınıflı verilerin sınıflandırılması için kullanılan OVO ve OVA stratejilerinin hem de bu stratejiler altında uygulanan bazı ikili sınıflandırıcıların performanslarının karşılaştırılmasıdır.

Çalışmanın Önemi

Yaşanılan teknolojik gelişmelerle beraber son yıllarda veri madenciliği uygulamaları her sahada gittikçe yaygınlaşmaktadır. Bununla beraber eğitim alanında veri madenciliği uygulamalarının kullanımının istenilen düzeyde olmadığı düşünülmektedir. Eğitim alanının her boyutundan toplanan verileri veri madenciliği yöntemleri ile modellemenin, daha etkili eğitim politikaları oluşturulması, öğrencilere ilişkin birçok sınıflandırma, tahminleme ve kümeleme işlemlerinin yapılması, başarısızlık-okul bırakma gibi durumların önceden tespiti ya da eğitim ve öğrenmeye ilişkin çeşitli kuramların oluşturulması gibi birçok durumda eğitim ve öğretim süreçlerine katkı sağlayabileceği ve verimliliği artırılabilirliği öngörülmektedir. Bu anlamda veri madenciliği yöntemlerinin eğitimde ölçme ve değerlendirme süreçlerine de yeni bir boyut kazandıracak potansiyele sahip olduğu söylenebilir. Tüm bu sebeplerle PISA 2015 Türkiye verileri ile yapılan bu çalışmanın hem algoritma performanslarının kıyaslanması hem de çok sınıflı sınıflandırma stratejilerinden olan OVA ve OVO stratejilerinin kullanılması bakımından önemli olduğu düşünülmektedir.

Yürütülen literatür taramasında Türkiye’de daha önce çok sınıflı eğitim verileri üzerinden ayırıştırma teknikleri kullanılarak yapılan herhangi bir çalışmaya rastlanmamıştır. Çok sınıflı verilere eğitimde oldukça sık rastlanmaktadır ve örnekleri doğru sınıflara dâhil edebilmek çok sınıflı veriler için bir problem olmaktadır. Bu anlamda çalışma, çok sınıflı eğitim verilerini sınıflandırmak için kullanılacak olan OVA ve OVO stratejilerini tanıtmakta ve bu stratejilerin eğitim verileri üzerinden bir uygulamasını içermektedir.

Çalışma bazı temel sınıflandırıcıların performanslarını değerlendirmek açısından da önemlidir. Algoritmaların örnekleri doğru sınıflandırma performansının yüksek olması değerlendirme yönteminin daha az hatalı olduğunu göstermektedir ki bu, etkili bir değerlendirme yapmak açısından oldukça önemlidir. Her ne kadar algoritmaların performansları kullanılan veri setine göre değişiklik gösterse de eğitim sektöründen gelen verilerle denenmesi çalışmanın önemi olarak değerlendirilebilecek bir husustur.

Problem

Bu çalışmanın ana problemi “Çok sınıflı verilerin sınıflandırılması için kullanılan çeşitli stratejiler altında, algoritmaların ve kullanılan stratejilerin performansları nasıldır?” şeklindedir.

Alt Problemler

- 1) OVA stratejisi altında doğruluk, ortalama hata, F istatistiği ve kappa istatistiği metrikleri bakımından en yüksek ve en düşük performans gösteren algoritmalar hangileridir?
- 2) OVA stratejisi altında en az sayıda veri içeren sınıftaki örnekleri doğru tahminleyebilme başarısı bakımından en iyi performans gösteren algoritma hangisidir?
- 3) OVO stratejisi altında doğruluk, ortalama hata, F istatistiği ve kappa istatistiği metrikleri bakımından en yüksek ve en düşük performans gösteren algoritmalar hangileridir?
- 4) OVO stratejisi altında en az sayıda veri içeren sınıftaki örnekleri doğru tahminleyebilme başarısı bakımından en iyi performans gösteren algoritma hangisidir?
- 5) Algoritmaların her birinin performansları OVA ve OVO stratejilerinden hangisinde daha yüksektir?
- 6) OVA ve OVO stratejilerinden hangisi çok sınıflı veri setini sınıflandırmada daha başarılıdır?

Yöntem

Bu bölümde çalışmada kullanılacak araştırma yöntemi, araştırma grubu, veri toplama araçları, uygulama süreci ve verilerin analizi ile ilgili bilgiler verilmiştir.

Araştırmanın Yöntemi ve Deseni

Çalışmada Uluslararası Öğrenci Değerlendirme Programı (PISA) fen başarısı bağımlı değişkenine göre sınıflandırma yapmak amacıyla OVA ve OVO ayırıştırma stratejileri çerçevesinde bazı veri madenciliği algoritmalarının performansları karşılaştırıldığından bu çalışma betimsel araştırma niteliindedir.

Çalışma Grubu

Millî Eğitim Bakanlığı Ölçme, Değerlendirme ve Sınav Hizmetleri Genel Müdürlüğü tarafından yürütülen PISA araştırmasının altıncı döngüsü olan 2015 uygulaması, Türkiye’de bilgisayar tabanlı olarak 61 ilden 187 okul ve 5895 öğrencinin katılımı ile gerçekleştirilmiştir. Okullar belirlenirken tabakalı seçkisiz örnekleme yöntemi kullanılmış daha sonra bu okullardan seçilen öğrenciler yine seçkisiz yöntemle belirlenmiştir. Okullar, istatistiki bölge birimleri sınıflamasına göre belirlenen 12 bölge içinden, eğitim türü, okul türü, okulların buldukları yer ve okulların idari biçimleri dikkate alınarak oluşturulmuştur (Millî Eğitim Bakanlığı [MEB], 2016).

Veri Toplama Araçları

PISA araştırması öğrencilerin hem “fen”, “matematik” ve “okuma” alanlarındaki okuryazarlıklarını belirlemek amacıyla geliştirilmiş başarı testlerini hem de öğrencilerin okul ve aile ortamlarını, motivasyonlarını, öğrenme süreçlerini değerlendirmek üzere geliştirilmiş anketleri içermektedir. Yapılan çalışmada fen başarı testinden elde edilen veriler ile anket sonuçlarından faydalanılmıştır. Her PISA döngüsünde değişmekle birlikte 2015 yılında ağırlıklı alan fen okuryazarlığı alanıdır. Bu sebeple öğrenci anketindeki maddeler de fen okuryazarlığına ilişkindir. Kullanılan değişkenlere Tablo 1’de yer verilmiştir.

Uygulama

5895 sayıda örnek içeren 2015 PISA Türkiye verilerine <https://www.oecd.org/pisa/data/2015database/> bağlantısı ile erişilerek içerisinden kayıp verileri silme ve atama işlemleri sonrası 3459 öğrenciye ait bir veri seti elde edilmiştir. Daha sonra bu veriler içerisinden belirlenen 26 bağımsız ve 1 bağımlı değişken kategorik olarak tanımlanmıştır. Bağımlı değişken olarak belirlenen fen okuryazarlığı değişkeni için ise 1 ve 2. düzeyler alt yeterlilik düzeyi, 3 ve 4. düzeyler orta yeterlilik düzeyi ve 5 ve 6. düzeyler ise üst yeterlilik düzeyi olarak değerlendirilmiştir. Elde edilen dosya arff formatına dönüştürülerek WEKA paket programında işlenmeye uygun hale getirilmiştir.

Verilerin Analizi

Verilerin analizinde WEKA paket programı içerisindeki Experimenter tezgahı kullanılmış, OVA ve OVO stratejileri altında belirlenen beş farklı algoritma veri setine uygulanmıştır. Çalışmada algoritmaların performanslarının birbiriyle kıyaslanabilecek şekilde olması ve yanlılık oluşmaması açısından parametre seçimi yapılmamış; tüm algoritmalar için varsayılan parametreler kullanılmıştır. Test seçeneklerinden 10 katlı çapraz geçirme, birleştirme stratejilerinden de oy verme tekniği kullanılmıştır.

Bulgular

Bu bölümde kullanılan veri setine ve analiz sonuçlarına ilişkin bulgular paylaşılmıştır. Veri setine ilişkin bilgilerde her bir tahmin edici değişkenin ve bağımlı değişken olan fen okuryazarlığının kategorilerindeki dağılımlar bir tablo ile verilmiştir. Tabloda rakamla belirtilen kategorilerin hangi değişkene karşılık geldiği tablonun altında belirtilmiştir.

Analiz sonuçlarına ilişkin bulgular çalışmanın ana ve alt problemlerine göre sınıflandırılarak verilmiştir. Bunun için OVA ve OVO stratejileri altında algoritmaların doğruluk, hata, F istatistiği ve Kappa istatistiği değerleri bir tablo ile gösterilmiştir. WEKA çıktıları olarak verilen şekillerde ise her bir algoritmanın yine OVA ve OVO stratejileri altında tüm sınıflar için gerçek pozitif oranı (TP rate), yanlış pozitif oranı (FP rate), kesinlik (precision), duyarlılık (recall),

F istatistiği ve Matthews Korelasyon Katsayısı (MCC) değerleri ile Alıcı İşlem Karakteristiği (ROC) ve Kesinlik-Duyarlılık Eğrisi (PRC) alanı değerleri verilmiştir. Ayrıca bu WEKA çıktılarında karışıklık matrisleri de bulunmaktadır. Son alt problem için J-48 algoritmasının ürettiği karar ağacı üzerinden fen okuryazarlığını etkileyen değişkenler yorumlanmıştır.

Tablo 1

Değişkenlerin Kategorilerine Göre Dağılımları

Değişkenler	1	2	3	4	5	6
Bölge	1130	458	571	515	270	515
Baba eğitim durumu	1251	902	503	803		
Anne eğitim durumu	1737	677	861	184		
Sınıf düzeyi	45	701	2675	38		
Cinsiyet	1736	1723				
Odaya sahip olma	2562	897				
Bilgisayara ulaşım imkanı	2468	991				
İnternete ulaşım imkanı	2298	1161				
Kaynak kitaplara ulaşım imkanı	2946	513				
Aile ilgisi	493	2966				
Okulla ilgili kaygı	1068	2391				
Hırslı olma	203	3256				
Takım çalışmasına yatkınlık	464	2995				
Olumsuz öğretmen davranışı	962	2497				
Fen sınıf disiplini	1121	2338				
Sorgulamaya dayalı öğretim	2853	606				
Okula ait hissetme	2791	668				
Öğretmen desteği	798	2661				
Öğretmen odaklı eğitim	2438	1021				
Geribildirim	3082	377				
Uyarlanabilir öğretim	1471	1988				
İçsel motivasyon	2720	739				
Araçsal motivasyon	1954	1505				
Özyeterlilik	2111	1348				
Epistemik inançlar	1814	1645				
Fen etkinliklerine katılım	808	2651				

Fen okuryazarlığı	2482	970	7
-------------------	------	-----	---

Tablo 1’de değişkenlerin kategorilerine göre dağılımları verilmiştir. Bölge değişkeni için 1: Marmara, 2: Ege, 3: İç Anadolu, 4: Akdeniz, 5: Karadeniz ve 6: Doğu ve Güneydoğu Anadolu Bölgeleri’ni göstermektedir. Baba eğitim durumu ve anne eğitim durumu değişkenleri için 1: okumamış ve ilkokul, 2: ortaokul, 3: lise, 4: üniversite ve üstü mezuniyet durumlarını göstermektedir. Sınıf düzeyi değişkeni için 1: 7. ve 8. sınıfları, 2: 9. sınıfları, 3: 10. sınıfları, 4 ise 11 ve 12. sınıfları göstermektedir. Cinsiyet değişkeninde 1: kızları 2 ise erkekleri göstermektedir. Kendine ait odaya sahip olma, bilgisayar, internet ve kaynak kitaplara ulaşım değişkenleri için 1: var, 2: yok anlamına gelmektedir. Aile ilgisi, sınav kaygısı, hırslı olma, takım çalışmasına yatkınlık, fen derslerindeki sınıf disiplini, olumsuz öğretmen davranışı, içsel motivasyon, epistemik inançlar ve okula ait hissetme değişkenlerinde 1: düşük 2: yüksek olma durumunu ifade etmekte iken araçsal motivasyon ve özyeterlilik değişkenlerinde 1: düşük 2 ise yüksek olma durumunu ifade etmektedir. Sorgulamaya dayalı öğretim ve öğretmen desteği değişkenleri için 1: derslerin çoğunda ya da her ders durumunu belirtirken, 2: bazen ya da hemen hemen hiç durumunu belirtmektedir. Öğretmen odaklı eğitim, geribildirim verme ve uyarlanabilir öğretim değişkenleri için ise 1: hiç ya da hemen hemen hiç durumunu ifade ederken; 2: derslerin çoğu ya da her ders durumunu ifade etmektedir. Fen etkinlikleri değişkeninde ise 1: düzenli aralıklarla ya da sık sık durumunu belirtirken; 2 ise hiç ya da bazen durumunu belirtmektedir. Bağımlı değişken durumunda olan fen okuryazarlığı değişkeninde ise 1b, 1a ve 2. Seviyede bulunan öğrenciler 1 koduyla; 3 ve 4. seviyedeki öğrenciler 2 koduyla ve 5 ve 6. Seviyedeki öğrenciler ise 3 koduyla verilmiştir.

Birinci ve İkinci Alt Probleme İlişkin Bulgular

OVA stratejisi altında algoritmaların doğruluk, hata ve Kappa istatistiği değerleri tablo 4’te gösterilmiştir.

Tablo 2

OVA Stratejisi Altında Algoritmaların Çeşitli Metrikler Açısından Performans Değerleri

Metrikler	Doğruluk	MAE	RAE	RMSE	RRSE	Kappa
Algoritmalar	(%)		(%)		(%)	
LR	74.1255	0.2221	81.9013	0.3365	91.4183	0.2750
NB	72.4776	0.2219	81.8183	0.3497	94.9900	0.3138
DVM	73.1714	0.1779	65.6084	0.4178	113.4935	0.1249
J-48	72.4198	0.2318	85.4799	0.3710	100.7778	0.2474
KNN	67.5629	0.2454	90.4998	0.4372	118.7806	0.1726

Tablo 2’ye göre doğruluk değeri en yüksek olan algoritma LR algoritmasıdır. Bu algoritmayı sırasıyla DVM, NB, J-48 ve KNN algoritmaları takip etmektedir. MAE ve RAE hata

değerleri bakımından algoritmalar DVM, NB, LR, J-48 ve KNN olarak sıralanmıştır. RMSE ve RRSE hata değerleri açısından ise algoritmalar LR, NB, J-48, DVM ve KNN olarak sıralanmışlardır. Kappa istatistiği metriği bakımından NB algoritmasını sırasıyla LR, J-48, KNN ve DVM algoritmaları takip etmektedir.

Algoritmaların hata oranlarını değerlendirebilmek için MAE ve RMSE değerlerinin ortalaması çok sık kullanılan bir göstergedir. Alternatif olarak, görel hatalar da kullanılabilir fakat ortalama değeri almak akıllıca olmaktadır (Bin Othman & Yau, 2007).

OVA stratejisi altında en yüksek ortalama hata değerine sahip algoritma 0.3413 değeriyle KNN algoritmasıdır. En düşük ortalama hata değerine sahip olan algoritma ise 0.2793 değeriyle LR algoritmasıdır. Aradaki algoritmalar ise sırasıyla 0.3014 değeriyle j-48 algoritması 2. Sırada, 0.2978 değeriyle SVM algoritması 3. Sırada ve 0.2858 değeriyle NB ise 4. Sırada.

OVA stratejisi altında WEKA çıktıları olarak verilen şekillerde ise her bir algoritmanın yine OVA stratejisi altında tüm sınıflar için gerçek pozitif oranı (TP rate), yanlış pozitif oranı (FP rate), kesinlik, duyarlılık, F ve MCC değerleri ile ROC ve PRC alanı değerleri verilmiştir. Ayrıca tüm şekiller algoritmaların her bir sınıftaki örnekleri nasıl yerleştirdiğini gösteren karışıklık matrislerini de içermektedir.

Şekil 1

OVA Stratejisi Altında LR Algoritmasının Her Sınıftaki Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC    ROC Area  PRC Area  Class
      0,904    0,656    0,778     0,904    0,836     0,299  0,762    0,889     1
      0,331    0,093    0,580     0,331    0,422     0,291  0,759    0,532     2
      0,000    0,006    0,000     0,000    0,000     -0,004 0,829    0,032     3
Weighted Avg. 0,741    0,497    0,721     0,741    0,718     0,297  0,762    0,787

=== Confusion Matrix ===

  a  b  c  <-- classified as
2243 228 11 |  a = 1
 638 321 11 |  b = 2
   3   4   0 |  c = 3
```

Şekil 1’de LR algoritmasının OVA stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerleri görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere LR algoritması 1.sınıfa ait olan örneklerin 2243 tanesini, 2.sınıfa ait olan örneklerin 321 tanesini doğru olarak sınıflandırmıştır. 3. sınıfa ait olan örneklerin hiçbirini doğru olarak sınıflandıramamıştır.

Şekil 2

OVA Stratejisi Altında NB Algoritmasının Bağımlı Değişkenin Sınıflarına Ait Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC    ROC Area  PRC Area  Class
      0,817    0,506    0,804    0,817    0,811    0,316  0,745    0,880    1
      0,493    0,182    0,513    0,493    0,503    0,314  0,743    0,509    2
      0,000    0,001    0,000    0,000    0,000    -0,002 0,901    0,035    3
Weighted Avg. 0,725    0,414    0,721    0,725    0,723    0,315  0,745    0,774

=== Confusion Matrix ===

  a  b  c  <-- classified as
2029 449 4 | a = 1
 492 478 0 | b = 2
   2   5 0 | c = 3
```

NB algoritmasının OVA stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerleri şekil 2’de görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere NB algoritması 1.sınıfa ait olan örneklerin 2029 tanesini, 2.sınıfa ait olan örneklerin 478 tanesini doğru olarak sınıflandırmıştır. 3. sınıfa ait olan örneklerin hiçbirini doğru olarak sınıflandıramamıştır.

Şekil 3

OVA Stratejisi Altında DVM Algoritmasının Bağımlı Değişkenin Sınıflarına Ait Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC    ROC Area  PRC Area  Class
      0,972    0,876    0,738    0,972    0,839    0,189  0,568    0,746    1
      0,123    0,029    0,623    0,123    0,205    0,184  0,567    0,340    2
      0,000    0,000    0,000    0,000    0,000    0,000  0,488    0,002    3
Weighted Avg. 0,732    0,637    0,704    0,732    0,659    0,187  0,567    0,631

=== Confusion Matrix ===

  a  b  c  <-- classified as
2412  70  0 | a = 1
  851 119  0 | b = 2
   5   2  0 | c = 3
```

DVM algoritmasının OVA stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerler şekil 3’te görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere DVM algoritması 1.sınıfa ait olan örneklerin 2412 tanesini, 2.sınıfa ait olan örneklerin 119 tanesini doğru olarak sınıflandırmıştır. 3. sınıfa ait olan örneklerin hiçbirini doğru olarak sınıflandıramamıştır.

Şekil 4

OVA Stratejisi Altında J-48 Algoritmasının Bağımlı Değişkenin Sınıflarına Ait Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
      0,873    0,638    0,777     0,873    0,822     0,268    0,618    0,769     1
      0,362    0,128    0,524     0,362    0,428     0,266    0,617    0,369     2
      0,000    0,000    0,000     0,000    0,000     0,000    0,500    0,002     3
Weighted Avg.  0,728    0,493    0,704     0,728    0,710     0,267    0,617    0,655

=== Confusion Matrix ===

  a   b   c  <-- classified as
2166 316   0 |   a = 1
  619 351   0 |   b = 2
    4   3   0 |   c = 3
```

J-48 algoritmasının OVA stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerleri şekil 4'te görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere J-48 algoritması 1.sınıfa ait olan örneklerin 2166 tanesini, 2.sınıfa ait olan örneklerin 351 tanesini doğru olarak sınıflandırmıştır. 3. sınıfa ait olan örneklerin hiçbirini doğru olarak sınıflandıramamıştır.

Şekil 5

OVA Stratejisi Altında KNN Algoritmasının Bağımlı Değişkenin Sınıflarına Ait Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
      0,694    0,509    0,776     0,694    0,733     0,174    0,594    0,759     1
      0,489    0,304    0,385     0,489    0,431     0,173    0,593    0,332     2
      0,286    0,002    0,222     0,286    0,250     0,250    0,711    0,078     3
Weighted Avg.  0,635    0,450    0,665     0,635    0,647     0,174    0,594    0,638

=== Confusion Matrix ===

  a   b   c  <-- classified as
1722 754   6 |   a = 1
  495 474   1 |   b = 2
    2   3   2 |   c = 3
```

KNN algoritmasının OVA stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerleri şekil 5'te görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere KNN algoritması 1.sınıfa ait olan örneklerin 1722 tanesini, 2.sınıfa ait olan örneklerin 474 tanesini, 3. sınıfa ait olan örneklerin ise 2 tanesini doğru olarak sınıflandırmıştır.

OVA stratejisi altında sadece KNN algoritması iki örneği doğru sınıfa dahil etmiştir. Diğer algoritmalar en az örneğe sahip olan sınıfın, yani 3 ile temsil edilen sınıfın, hiçbir örneğini doğru sınıfa yerleştirememişlerdir.

Tablo 3

OVA Stratejisi Altında Algoritmaların Karşılaştırılması

	Doğruluk	Kesinlik	Duyarlılık	F istatistiği
LR	74.1255	0.7210	0.7410	0.7180
NB	72.4776	0.7210	0.7250	0.7230
DVM	73.1714	0.7040	0.7320	0.6590
J-48	72.4198	0.7040	0.7280	0.7100
KNN	67.5629	0.6660	0.6350	0.6470

Tablo 3'teki veriler incelendiğinde OVA stratejisi altında %74.1255 doğruluk değeriyle LR algoritmasının en iyi sonucu ürettiği söylenebilir. Algoritmaların performanslarının değerlendirilmesinde doğruluk ölçütü basit fakat önemli bir kriterdir. Kesinlik ölçütü bakımından 0.721 değeriyle en yüksek performansı gösteren algoritmalar NB ve LR algoritmaları olsa da kesinlik ölçütünün tek başına yorumlanması yanlış değerlendirmelere sebep olabilir. Bu sebeple kesinlik ölçütü duyarlılık ölçütüyle birlikte değerlendirilmelidir. Duyarlılık ölçütüne göre en başarılı algoritma 0.741 değeriyle LR algoritmasıdır. Kesinlik ve duyarlılık metriklerinin bir arada değerlendirilmesi için bu metriklerin harmonik ortalaması olan F istatistiği göz önüne alınmalıdır. F istatistiğine göre ise 0.723 değeriyle en başarılı algoritma NB algoritmasıdır.

Üçüncü ve Dördüncü Alt Probleme İlişkin Bulgular

OVO stratejisi altında tüm algoritmaların doğruluk ve Kappa istatistiği değerleri ile hata değerleri tablo 4'te verilmiştir.

Tablo 4

OVO Stratejisi Altında Doğruluk, Hata ve Kappa İstatistiği Değerleri

Metrikler	Doğruluk	MAE	RAE	RMSE	RRSE	Kappa
Algoritmalar	(%)		(%)		(%)	
LR	74.2122	0.2809	103.6096	0.3641	98.9187	0.2726
NB	72.3041	0.2846	104.9601	0.3694	100.3543	0.3110
DVM	73.3738	0.2818	103.9413	0.3656	99.3333	0.1460
J-48	72.7667	0.2832	104.4389	0.3675	99.8332	0.2582
KNN	63.5444	0.3039	112.0681	0.3946	107.2075	0.1712

OVO stratejisi altında algoritmaların çeşitli metrikler açısından değerleri tabloda verilmiştir. Tablo 4'e göre doğruluk değeri en yüksek olan algoritma LR algoritmasıdır. Bu algoritmayı sırasıyla DVM, NB, J-48 ve KNN algoritmaları takip etmektedir. MAE hata metriği bakımından LR algoritmasını sırasıyla DVM, NB, J-48 ve KNN algoritmaları takip etmektedir. RAE, RMSE ve RRSE metrikleri bakımından ise algoritmalar LR, DVM, J-48, NB, KNN olarak sıralanmaktadır. Kappa istatistiği ölçütüne göre ise sıralama NB, LR, J-48, KNN ve DVM şeklindedir.

Algoritmaların hata oranlarını değerlendirebilmek için MAE ve RMSE ortalaması çok sık kullanılan bir göstergiyi ifade etmektedir. Alternatif olarak, görelî hatalar da kullanılır. Fakat ortalama değeri almak akıllıca olmaktadır. Düşük hata değerine sahip olan algoritma daha güçlü sınıflandırma yeteneğine sahip olduğundan tercih edilmektedir (Bin Othman & Yau, 2007).

OVO stratejisi altında en yüksek ortalama hata değerine sahip algoritma 0.3492 değeriyle KNN algoritmasıdır. En düşük ortalama hata değerine sahip olan algoritma ise 0.3225 değeriyle LR algoritmasıdır. Aradaki algoritmalar ise sırasıyla 0.327 değeriyle NB algoritması 2. Sırada, 0.3253 değeriyle J-48 algoritması 3. Sırada ve 0.3237 değeriyle DVM ise 4. sırada olacak şekilde sıralanmışlardır.

Şekil 6

OVO Stratejisi Altında LR Algoritmasının Bağımlı Değişkenin Sınıflarına Ait Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
      0,906   0,664   0,776     0,906   0,836     0,295   0,624    0,772     1
      0,328   0,093   0,579     0,328   0,419     0,289   0,617    0,379     2
      0,143   0,003   0,077     0,143   0,100     0,102   0,705    0,024     3
Weighted Avg.  0,742   0,503   0,719     0,742   0,717     0,293   0,622    0,660

=== Confusion Matrix ===

  a  b  c  <-- Classified as
2248 228 6 |  a = 1
 646 318 6 |  b = 2
   3   3 1 |  c = 3
```

LR algoritmasının OVO stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerleri şekil 6'da görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere LR algoritması 1.sınıfa ait olan örneklerin 2248 tanesini, 2.sınıfa ait olan örneklerin 318 tanesini, 3. Sınıfa ait olan örneklerin ise 1 tanesini doğru olarak sınıflandırmıştır.

Şekil 7

OVO Stratejisi Altında NB Algoritmasının Bağımlı Değişkenin Sınıflarına Ait Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
      0,816    0,506    0,804     0,816    0,810     0,314    0,656    0,789     1
      0,490    0,182    0,512     0,490    0,501     0,312    0,654    0,394     2
      0,000    0,003    0,000     0,000    0,000     -0,003   0,566    0,005     3
Weighted Avg.  0,723    0,414    0,720     0,723    0,722     0,313    0,655    0,676

=== Confusion Matrix ===

  a  b  c  <-- classified as
2026 448 8 |  a = 1
492 475 3 |  b = 2
  2  5  0 |  c = 3
```

NB algoritmasının OVO stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerleri şekil 7’de görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere NB algoritması 1.sınıfa ait olan örneklerin 2026 tanesini, 2.sınıfa ait olan örneklerin 475 tanesini doğru olarak sınıflandırmıştır. 3. Sınıfa ait olan örneklerin hiçbirini doğru olarak sınıflandıramamıştır.

Şekil 8

OVO Stratejisi Altında DVM Algoritmasının Bağımlı Değişkenin Sınıflarına Ait Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
      0,965    0,852    0,742     0,965    0,839     0,204    0,557    0,741     1
      0,147    0,036    0,616     0,147    0,238     0,201    0,556    0,330     2
      0,000    0,000    0,000     0,000    0,000     0,000    0,500    0,002     3
Weighted Avg.  0,734    0,621    0,705     0,734    0,669     0,203    0,556    0,624

=== Confusion Matrix ===

  a  b  c  <-- classified as
2395 87  0 |  a = 1
827 143 0 |  b = 2
  5  2  0 |  c = 3
```

DVM algoritmasının OVO stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerleri şekil 8’de görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere DVM algoritması 1.sınıfa ait olan örneklerin 2395 tanesini, 2.sınıfa ait olan örneklerin 143 tanesini doğru olarak sınıflandırmıştır. 3. Sınıfa ait olan örneklerin hiçbirini doğru olarak sınıflandıramamıştır.

Şekil 9

OVO Stratejisi Altında J-48 Algoritmasının Bağımlı Değişkenin Sınıflarına Ait Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
      0,873    0,638    0,777     0,873   0,822     0,268    0,618    0,769     1
      0,362    0,128    0,524     0,362   0,428     0,266    0,617    0,369     2
      0,000    0,000    0,000     0,000   0,000     0,000    0,500    0,002     3
Weighted Avg.  0,728    0,493    0,704     0,728   0,710     0,267    0,617    0,655

=== Confusion Matrix ===

  a   b   c  <-- classified as
2166 316   0 |   a = 1
 619 351   0 |   b = 2
   4   3   0 |   c = 3
```

J-48 algoritmasının OVO stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerler şekil 9’da görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere J-48 algoritması 1.sınıfa ait olan örneklerin 2166 tanesini, 2.sınıfa ait olan örneklerin 351 tanesini doğru olarak sınıflandırmıştır. 3. Sınıfa ait olan örneklerin hiçbirini doğru olarak sınıflandıramamıştır.

Şekil 10

OVO Stratejisi Altında KNN Algoritmasının Bağımlı Değişkenin Sınıflarına İlişkin Performans Değerleri ve Karışıklık Matrisi

```
=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
      0,694    0,509    0,776     0,694   0,733     0,174    0,594    0,759     1
      0,489    0,304    0,385     0,489   0,431     0,173    0,593    0,332     2
      0,286    0,002    0,222     0,286   0,250     0,250    0,711    0,078     3
Weighted Avg.  0,635    0,450    0,665     0,635   0,647     0,174    0,594    0,638

=== Confusion Matrix ===

  a   b   c  <-- classified as
1722 754   6 |   a = 1
 495 474   1 |   b = 2
   2   3   2 |   c = 3
```

KNN algoritmasının OVO stratejisi altında bağımlı değişkenin her bir kategorisine ilişkin değerler şekil 10’da görülmektedir. Ayrıca karışıklık matrisinde de görüldüğü üzere KNN algoritması 1.sınıfa ait olan örneklerin 1722 tanesini, 2.sınıfa ait olan örneklerin 474 tanesini, 3. Sınıfa ait olan örneklerin ise 2 tanesini doğru olarak sınıflandırmıştır.

OVO stratejisi altında en az örnekli sınıf için LR algoritması 1 örneği, KNN algoritması ise 2 örneği doğru sınıfa dâhil etmiştir. Diğer algoritmalar ise hiçbir örneğin sınıfını doğru tahminleyememişlerdir.

Tablo 5

OVO Stratejisi Altında Algoritmaların Doğruluk, Kesinlik, Duyarlılık ve F İstatistiği Değerleri

	Doğruluk	Kesinlik	Duyarlılık	F İstatistiği
LR	74.2122	0.7190	0.7420	0.7170
NB	72.3041	0.7200	0.7230	0.7220
DVM	73.3738	0.7050	0.7340	0.6690
J-48	72.7667	0.7040	0.7280	0.7100
KNN	63.5444	0.6650	0.6350	0.6470

Tablo 5'teki veriler incelendiğinde OVO stratejisi altında %74.2122 doğruluk derecesiyle LR algoritmasının en iyi sonucu ürettiği söylenebilir. Algoritmaların performanslarının değerlendirilmesinde doğruluk ölçütü basit fakat önemli bir kriterdir. Kesinlik ölçütü bakımından 0.720 değeriyle en yüksek performansı gösteren algoritma NB olsa da kesinlik ölçütünün tek başına yorumlanması yanlış değerlendirmelere sebep olabilir. Kesinlik ölçütü duyarlılık ölçütüyle birlikte değerlendirilmelidir. Duyarlılık ölçütüne göre en başarılı algoritma 0.742 değeriyle LR algoritmasıdır. Kesinlik ve duyarlılık metriklerinin bir arada değerlendirilmesi için bu metriklerin harmonik ortalaması olan F istatistiğine göre en başarılı algoritma NB algoritmasıdır.

Beşinci ve Altıncı Alt Probleme İlişkin Bulgular

Algoritmaların her birinin OVA ve OVO stratejileri altındaki tüm metriklerdeki performans değerleri karşılaştırmalı olarak tablo 6'da verilmiştir. Ayrıca OVA ve OVO stratejilerinden hangisinin veri setini sınıflandırmada daha başarılı olduğu da tablo 8 yardımıyla değerlendirilecektir.

Tablo 6

Algoritmaların Stratejiler Bazında Karşılaştırılması

	LR		NB		DVM		J-48		KNN		Ortalama	
	OVA	OVO	OVA	OVO	OVA	OVO	OVA	OVO	OVA	OVO	OVA	OVO
Doğruluk	74.1255	74.2122	72.4776	72.3041	73.1714	73.3738	72.4198	72.7667	67.5629	63.5444	71.9514	71.2402

Kesinlik	0.721	0.719	0.721	0.720	0.704	0.705	0.704	0.704	0.666	0.665	0.7032	0.7026
Duyarlılık	0.741	0.742	0.725	0.723	0.732	0.734	0.728	0.728	0.635	0.635	0.7122	0.7124
F istatistiği	0.718	0.717	0.723	0.722	0.659	0.669	0.710	0.710	0.647	0.647	0.6914	0.693
Kappa	0.275	0.2726	0.3138	0.311	0.1248	0.146	0.2474	0.2582	0.1726	0.1712	0.2267	0.2318

LR algoritmasının doğruluk ve duyarlılık değerleri OVO stratejisinde daha yüksek iken kesinlik, F istatistiği ve kappa istatistiği değerleri OVA stratejisinde daha yüksektir. NB algoritmasının tüm değerleri OVA stratejisinde daha yüksektir. DVM algoritmasının tüm değerleri OVO stratejisi altında daha yüksektir. J-48 algoritmasının doğruluk ve kappa istatistiği değerleri OVO stratejisi altında daha yüksek iken diğer değerleri her iki stratejide de aynı kalmıştır. KNN algoritmasının doğruluk, kesinlik ve kappa istatistiği değerleri OVA stratejisi altında daha yüksektir, duyarlılık ve F değerleri ise her iki strateji altında da eşittir. Ortalama değerler incelendiğinde doğruluk ve kesinlik değerleri bakımından OVA stratejisi, duyarlılık, F istatistiği ve kappa istatistiği metrikleri bakımından ise OVO stratejisinin başarı ortalamasının daha yüksek olduğu söylenebilir.

Doğruluk değeri bakımından her iki stratejiyi ve tüm algoritmaları göz önüne alırsak en başarılı algoritma OVO stratejisi altında LR algoritmasıdır. Kesinlik değerlerine bakıldığında OVA stratejisi altında LR ve NB algoritmaları en başarılı algoritmalarıdır. Duyarlılık değerleri bakımından OVO stratejisi altında LR algoritması en başarılı algoritma olarak değerlendirilebilir. F istatistiği bakımından ise OVA stratejisi altında NB algoritmasının en başarılı algoritma olduğu söylenebilir.

Sonuç ve Tartışma

Bu bölümde öncelikle çalışmaya ilişkin sonuçlar alt problemler çerçevesinde ortaya konularak ve sonuçlara ilişkin yorumlar yapılarak literatürden çeşitli çalışmalarla desteklenmiştir.

Birinci ve ikinci alt problemlere ilişkin sonuçlar değerlendirildiğinde; OVA stratejisi altında doğruluk ve ortalama hata değerleri bakımından LR algoritmasının, F istatistiği bakımından ise NB algoritmasının en başarılı algoritmalar olduğu görülmektedir. Her üç metrik bakımından da en düşük performansı gösteren algoritma KNN algoritmasıdır. En az sayıda örnek içeren sınıftaki yedi örnekten iki tanesini doğru sınıflandıran KNN algoritması bu değerlendirme ölçüsü bakımından en başarılı algoritmadır. Diğer dört algoritma bu sınıfa ait hiçbir örneği doğru

sınıflandıramamıştır. Kappa istatistiği metriği bakımından OVA stratejisi altında en başarılı algoritmanın NB algoritması olduğu görülmektedir. Bu algoritmayı LR algoritması izlemektedir.

Üçüncü ve dördüncü alt probleme ilişkin sonuçlar değerlendirildiğinde; OVO stratejisi altında doğruluk ve ortalama hata metrikleri bakımından LR algoritmasının ve F metriği bakımından ise NB algoritmasının en yüksek performansa sahip olduğu görülmektedir. En düşük performansa sahip olan algoritma ise her üç metrik açısından da KNN algoritmasıdır. En az sayıda örnek içeren sınıftaki yedi örnekten iki tanesini doğru sınıflandıran KNN algoritması ve bu örneklerden bir tanesini doğru olarak sınıflandıran LR algoritması diğer algoritmalarla kıyaslandığında en başarılı algoritmalar olarak değerlendirilebilir. Kappa istatistiği ölçüsü bakımından değerlendirildiğinde ise NB algoritması ve bu algoritmayı takiben LR algoritması en yüksek performansı sergilemiştir.

İlk dört alt probleme ilişkin sonuçlar birlikte değerlendirildiğinde doğruluk, hata ve kappa istatistiği metriklerinin her üçü bakımından da, hem OVA hem de OVO stratejisi altında en başarılı algoritmaların LR ve NB algoritmaları olduğu görülmektedir. Veri setinin dengesiz olması, bağımlı değişkenin üç sınıf olarak bölünmesi ve veri seti ile ilgili diğer parametreler bu sonucu ortaya çıkarmış olabilir. Her iki strateji altında da en düşük performansa sahip algoritmanın KNN algoritması olduğu da görülmektedir. Yapılan çalışmada KNN algoritmasında örnekler arası uzaklıkların öklit uzaklığına bağlı olarak belirlenmesi ve veri setinde çoğunluğu oluşturan örneklerin birbirine yakın bir grup oluşturmaması, bu algoritmanın başarısının düşük olmasının sebebi olabilir. Buna karşılık en az örnekli sınıfı tahmin başarısı bakımından KNN algoritmasının nispeten daha başarılı olduğu da görülmektedir. En az örnek içeren sınıftaki örneklerin bağımsız değişkenlere ait değerlerinin dağılımı birbirine yakındır. Bu sebeple öklit uzaklığı bu sınıf için avantajlı bir parametre olmaktadır. Altman (1990)'a göre her iki strateji altında da DVM ve KNN algoritmalarının kappa istatistiği değerleri 0.20 değerinin altındadır ve bu sonuçlar zayıf olarak değerlendirilir. Diğer algoritmaların kappa istatistiği değerleri 0.20 ile 0.40 arasında değiştiğinden makul düzeyde olarak değerlendirilir. Bunun sebebi olarak veri setinin dengesiz olması gösterilebileceği gibi algoritmanın formülasyonu da gösterilebilir. Algoritmalar daha fazla sayıda örnek içeren sınıflara bağlı olarak öğrenme gerçekleştirdiklerinden, çalışmada kullanılan algoritmaların kappa istatistiği değerleri düşük çıkmış olabilir. En az sayıda veri içeren sınıfın doğru tahmin edilme performanslarının düşük olması da bu savı güçlendirmektedir. OVO stratejisi az örnekli sınıflarla karşılaştığında zorlanmakta ve algoritmalar ezberlemeye yatkın hale gelmektedir. OVA stratejisinde ise bir sınıf, diğer tüm sınıfların birleşmesiyle oluşturulan gruba karşı indüklendiğinden sınıf dengesizliğinin negatif etkisi artmaktadır. Bu durum her iki stratejide de hem algoritmaların performanslarını düşürmüş hem de en az örnekli sınıfın doğru tahminlenme oranını etkilemiştir.

Beşinci ve altıncı alt problemlere ilişkin sonuçlar incelendiğinde, ortalama hata değeri bakımından tüm algoritmalar OVA stratejisi altında daha başarılıdır. DVM algoritması doğruluk ve F metriklerinin her ikisi bakımından da OVO stratejisi altında daha başarılı iken NB algoritması doğruluk ve F değerleri bakımından OVA stratejisi altında daha başarılıdır. LR algoritması doğruluk değeri bakımından OVO stratejisi altında, F istatistiği bakımından OVA stratejisi altında daha başarılıdır. J-48 algoritması doğruluk değeri bakımından OVO stratejisi altında daha başarılı iken F metriği bakımından her iki stratejide de performansının aynı olduğu görülmüştür. KNN algoritması doğruluk değeri bakımından OVA stratejisi altında daha başarılı iken F metriği bakımından da her iki stratejide de performansının aynı olduğu gözlenmiştir. Algoritmalar

doğruluk değeri ve hata ortalaması bakımından OVA stratejisi altında F metriği bakımından ise OVO stratejisi altında daha başarılıdır. Kappa istatistiği metriği bakımından LR, NB ve KNN algoritmaları OVA stratejisi altında DVM ve J-48 algoritmaları OVO stratejisi altında daha başarılıdır. Bu metrik bakımından stratejilerin ortalama değerlerine bakıldığında ise OVO stratejisinin daha başarılı olduğu görülmektedir.

Literatürde algoritmaların ve stratejilerin karşılaştırılması amacıyla birçok çalışma yapılmıştır. Kullanılan veri seti, seçilen parametreler, karşılaştırılan teknik ve algoritmalar farklı olduğundan sonuçlar çeşitlilik göstermektedir. Bu çalışmaya paralel sonuçlar ortaya çıktığı gibi farklı sonuçlar ortaya koyan çalışmalara da ulaşılmıştır. Ulaşılan çalışmalardan Bulut (2016), dengesiz veri setleri üzerinde algoritmaları karşılaştırmak üzere yaptığı çalışmada LR algoritmasını en yüksek performansa sahip olan algoritma olarak değerlendirmiştir. Çalışmasında 13 veri seti kullanmıştır ve LR algoritması bu 13 veri setinin 8'inde AUC metriği bakımından en yüksek performansı sergilemiştir. Çalışmada kullandığı diğer algoritmalar ise J-48, Naive Bayes, KNN ve DVM algoritmalarıdır. Bu çalışmada KNN ve DVM algoritmaları ise sadece birer veri setinde en yüksek performans göstermiş, en başarısız algoritmalar olarak değerlendirilmişlerdir. Sabzevari vd. (2007) kredi skorlaması tahmini üzerine yaptıkları araştırmada istatistiksel tekniklerden LR algoritmasını, veri madenciliği tekniklerinden ise Bagging algoritmasını sınıflandırma doğruluğu metriği bakımından en başarılı bulmuşlardır. Sharma ve Sahni (2011) yürüttükleri çalışmada, çeşitli karar ağacı algoritmalarını karşılaştırmışlar ve doğruluk metriği bakımından en başarılı karar ağacı algoritmasının J48 algoritması olduğunu gözlemişlerdir. Coşkun (2010); NB, LR, K-Star ve J-48 algoritmalarını karşılaştırdığı çalışmasında doğruluk ve F metrikleri bakımından en başarılı algoritmanın J-48 algoritması olduğunu belirtmiştir. Hassan vd. (2016) yaptıkları araştırmada, içinde LR, K-star ve J-48 algoritmalarının da bulunduğu 10 farklı algoritmadan Bayes Network algoritmasını, gerçek pozitif oranı, yanlış pozitif oranı, kesinlik, duyarlılık, F istatistiği, ROC alanı ve hesaplama süresi gibi birçok metrik bakımından en başarılı bulmuşlardır. Kahraman vd. (2004), YSA ile DVM algoritmalarını karşılaştırdıkları çalışmalarında sınıflandırma başarısı ve hesaplama süresi ölçüleri bakımından DVM algoritmasının YSA algoritmasından çok daha yüksek performans gösterdiğini belirtmişlerdir. Akçapınar (2014) yürüttüğü çalışmada, veri setindeki değişkenlerin ifade edilmiş türlerini değiştirerek (kategorik, sürekli gibi) ve özellik seçme yöntemlerini kullanarak birçok sınıma yapmış, tüm sınamalarda KNN algoritmasının tüm metrikler bakımından başarılı performansa sahip olduğunu gözlemlemiştir. Yurdakul (2015) çalışmasında J-Rip, KNN, J-48 ve NB algoritmalarının performanslarını doğruluk metriğine göre kıyaslamış ve en düşük performans gösteren algoritmanın NB algoritması olduğunu ifade etmiştir. Cong vd. (2009) çalışmalarında romatoid artrit hastalığı ile ilgili olduğu düşünülen bir enzimi belirlemek üzere makine öğrenmesi algoritmalarının performanslarını sınamış ve KNN algoritması %98.32 başarı performansı ile en başarılı algoritma olarak belirlemiştir. Varpa vd. (2011), KNN ve DVM algoritmalarını OVO ve OVA stratejileri altında kıyasladıkları çalışmalarında OVO stratejisinin OVA stratejisinden daha başarılı olduğunu ayrıca OVO+KNN birleşiminin en yüksek performansa sahip olduğunu tespit etmişlerdir.

Algoritmaların performanslarının karşılaştırıldığı araştırmalar sonuçları itibariyle çok çeşitlilik göstermektedir. Kullanılan veri seti, karşılaştırılan algoritmalar ve seçilen parametreler farklı olduğundan bu çeşitlilik oldukça normal bir durum olarak değerlendirilmektedir. Fakat bu farklılıklardan dolayı bu çalışmaları eleştiren araştırmacılar da bulunmaktadır. Hand (2006) parametreler, veri seti üzerinde yapılan ön işlemler gibi birçok durumun farklı olması sebebiyle bu tür karşılaştırma çalışmalarının illüzyon yarattığını ifade etmiştir. Ayrıca geliştirilen yeni

algoritmaların daha başarılı performans göstermesi adına yanlış davranılması ve bu çalışmalarda kullanılan veri setlerinin gerçek veri seti olmaması, performans karşılaştırma çalışmalarının eleştirilmesine sebep olmaktadır.

Fakat tüm bu eleştirilere rağmen algoritmaların karşılaştırılmasına yönelik yapılan çalışmalar literatürde kabul görmüştür ve çalışılan bir alan olmaya devam etmektedir. Veri seti-başarılı algoritma şeklinde ikililerin oluşabileceği düşüncesi bu tür çalışmaların üretilmesine sebep olmakta ve performans karşılaştırma çalışmalarının zamanla veri seti-başarılı algoritma eşleşmelerini belirginleştireceği düşünülmektedir.

Öneriler

1) Aynı veri seti üzerinde farklı algoritmaların (Bagging, boosting, diğer bayesyen, farklı karar ağacı vb.) performansları kıyaslanabilir.

2) Benzer bir çalışma anketle elde edilen eğitim verileri, ÖSYM'nin yaptığı sınavlar, uzaktan eğitim sistemleri veya farklı uluslararası sınavlar (TIMMS, PIRLS vb.) üzerinden toplanan veri setleri ile tekrarlanabilir.

3) Bu çalışmada veri madenciliği alanında en çok kullanılan programlardan biri olan Weka ile çalışılmıştır. Bundan sonra yapılacak çalışmalarda farklı veri madenciliği programları kullanılarak elde edilen sonuçlar karşılaştırılabilir.

4) Eğitim fakültesindeki öğrencilerin Türkçe, matematik ve fen derslerindeki başarılarını yordamak amacıyla okulun veri tabanından yararlanılarak; veri madenciliğine dayalı çalışmalar yapılabilir.

5) Elde edilen sonuçları sadece karışıklık matrisinden elde edilen değerlere göre yorumlamak yerine AUC metriği gibi farklı performans metriklerinden yararlanılabilir.

6) Araştırmada kullanılan OVA ve OVO stratejileri dışındaki farklı ayrıştırma stratejileri ve majority voting (çoğunluk oylaması) yöntemi dışındaki sonuçları birleştirme stratejileri dışında başka stratejiler kullanılabilir.

7) Çalışmada test seçeneklerinden 10 katlı çapraz geçirme yöntemi kullanılmıştır. Bundan sonraki çalışmalarda holdout metodu gibi farklı test seçenekleri kullanılabilir.

8) Çalışmada kullanılan algoritmalar değiştirilebileceği gibi algoritmaların parametreleri de değiştirilebilir. Örneğin KNN algoritmasında uzaklık, Öklit uzaklığı ile değil de Manhattan uzaklığı gibi kullanılacak farklı yöntemlerle çalışma tekrarlanabilir. Ya da DVM algoritmasında farklı çekirdek fonksiyonları kullanılabilir.

9) Veri seti üzerinde yapılacak farklı ön işlemler ile çalışma tekrarlanabilir. Örneğin oversampling, undersampling gibi verileri dengelemek için geliştirilen yöntemler kullanılabilir. Boş verilerin silinmesi yerine çeşitli yöntemlerle (örneğin karar ağaçları) atama yapılabilir. Değişkenler farklı türlerde düzenlenerek (örneğin kategorik değil de nicel bırakılarak) analiz tekrarlanabilir. Daha çok değişken veri setine dâhil edilebilir ya da, özellik seçimi yöntemleri ile değişkenler azaltılarak denenebilir.

Etik Kurul İzin Bilgisi: Yapılan bu çalışmada “Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesi” ’nde uyulması belirtilen tüm kurallara uyulmakla birlikte yönergenin ikinci bölümü olan “Bilimsel Araştırma ve Yayın Etiğine Aykırı Eylemler” başlığı altında belirtilen

eylemlerden hiçbiri gerçekleştirilmemiştir. Çalışmada, OECD'nin topladığı ve tüm araştırmacılara sunduğu PISA verileri üzerinde analizler yapıldığından etik kurul izni alınmasına gerek görülmemiştir.

Yazar Çıkar Çatışması Bilgisi: Bu çalışmada çıkar çatışması yoktur ve finansman desteği alınmamıştır.

Yazar Katkısı: Yazarlar makaleye eşit katkı sağlamış olduklarını beyan ederler.

Kaynakça

- Adnan, M. N., & Islam, M. Z. (2015, April 22-24). *One-vs-all binarization technique in the context of random forest* [Paper presentation]. European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), Bruges, Belgium.
- Akçapınar, G. (2014). *Çevrimiçi öğrenme ortamındaki etkileşim verilerine göre öğrencilerin akademik performanslarının veri madenciliği yaklaşımı ile modellenmesi* [Yayımlanmış doktora tezi, Hacettepe Üniversitesi]. YÖK Tez Merkezi.
- Akpınar, H. (2000). Veri tabanlarında bilgi keşfi ve veri madenciliği. *İstanbul Üniversitesi İşletme Fakültesi Dergisi*, 29(1), 1-22.
- Allwein, E. L., Schapire, R. E., & Singer, Y. (2000). Reducing multiclass to binary: A unifying approach for margin classifiers. *Journal of Machine Learning Research*, 1, 113-141.
- Altman, D. G. (1990). *Practical statistics for medical research*. CRC press.
- Aydoğan, E. (2008). *Veri madenciliğinde sınıflandırma problemleri için evrimsel algoritma tabanlı yeni bir yaklaşım: Rough-Mep algoritması* [Yayımlanmamış doktora tezi]. Gazi Üniversitesi.
- Babadağ, K. (2006). Zeki veri madenciliği: Ham veriden altın bilgiye ulaşma yöntemleri. *Industrial Application Software*, 85-87.
- Batista, G. E. A. P. A., & Silva, D. F. (2009, August 24-25). *How k-nearest neighbor parameters affect its performance* [Paper presentation]. 10th Argentine Symposium on Artificial Intelligence (ASAI 2009), Mar Del Plata, Argentina.
- Bhatia, N., & Vandana (2010). Survey of nearest neighbor techniques. *International Journal of Computer Science and Information Security*, 8(2), 302-305.
- Bin Othman, M. F., & Yau, T. M. S. (2007, December 11-14). *Comparison of different classification techniques using WEKA for breast cancer* [Paper presentation]. 3rd Kuala Lumpur International Conference on Biomedical Engineering, Kuala Lumpur, Malaysia.
- Bulut, F. (2016, April 26-27). *Performance evaluations of supervised learners on imbalanced datasets* [Paper presentation]. In 2016 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT), Istanbul, Turkey.
- Cong, Y., Yang X. G., Lv, W., & Xue, Y. (2009). Prediction of novel and selective TNF-alpha converting enzyme (TACE) inhibitors and characterization of correlative molecular descriptors by machine learning approaches. *Journal of Molecular Graphics and Modelling*, 28(3), 236-244.

- Coomans, D., & Massart, D. L. (1982). Alternative k-nearest neighbour rules in supervised pattern recognition: Part 1. K-nearest neighbour classification by using alternative voting rules. *Analytica Chimica Acta*, 136, 15-27.
- Coşkun, C. (2010). *Veri madenciliği algoritmalarının karşılaştırılması* [Yayımlanmış yüksek lisans tezi, Dicle Üniversitesi]. YÖK Tez Merkezi.
- Coşkun, C., & Baykal, A. (2011). *Veri madenciliğinde sınıflandırma algoritmalarının bir örnek üzerinde karşılaştırılması*. Akademik Bilişim.
- Cover, T. M., & Hart, P. E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21–27.
- Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. (Eds.) (1996). *Advances in knowledge discovery and data mining*. MIT Press/AAAI Press.
- Field, A. (2005). *Discovering statistics using SPSS* (2nd Edition). Sage Publications.
- Friedman, J. (1996). *Another approach to polychotomous classification*. Technical report, Department of Statistics, Stanford University, Stanford, CA.
- Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2011). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4), 463-484.
- Galar, M., Fernández, A., Barrenechea, E., & Herrera, F. (2015). DRCW-OVO: distance-based relative competence weighting combination for one-vs-one strategy in multi-class problems. *Pattern Recognition*, 48(1), 28-42.
- Hair, J. F., Black, W. C., Babin, B., Anderson, R. E., & Tatham, R. L. (2006). *Multivariate data analysis* (6th Edition). Prentice-Hall.
- Han, J., Kamber, M., & Pei, J. (2011). *Data mining: Concepts and techniques* (3rd Edition). Morgan Kaufmann Publishers.
- Hand, D. J. (2006). Classifier technology and the illusion of progress. *Institute of Mathematical Statistics in Statistical Science*, 21(1), 1-15.
- Hung, S. Y., Yen, D. C., & Wang, H. Y. (2006). Applying data mining to telecom churn management. *Expert Systems with Applications*, 31, 515–524.
- International Educational Data Mining Society [IEDMS], (2016). Educational data mining. 26 Aralık 2017 <http://www.educationaldatamining.org/> adresinden erişildi.
- Kahraman, F., Çapar, A., Ayvacı, A., Demirel, H., & Gökmen, M. (2004, April 30). *Comparison of DVM and ANN performance for handwritten character classification* [Paper presentation]. Proceedings of the IEEE 12th Signal Processing and Communications Applications Conference, Kusadasi, Turkey
- Kalıpsız, O., & Cihan, P. (2015). Öğrenci proje anketlerini sınıflandırmada en iyi algoritmanın belirlenmesi. *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi*, 8(1), 41-49.

- Kayri, M., & Çokluk, Ö. (2010). Using multinomial logistic regression analysis in artificial neural network: An application. *Ozean Journal of Applied Sciences, 3(2)*, 259-268.
- Liu, B., Hao, Z., & Tsang, E. C. (2008). Nesting one-against-one algorithm based on DVMS for pattern classification. *IEEE Transactions on Neural Networks, 19(12)*, 2044-2052.
- Lorena, A. C., De Carvalho, A. C., & Gama, J. M. P. (2008). A review on the combination of binary classifiers in multiclass problems. *Artificial Intelligence Review, 30(1)*, 19-37.
- Mertler, C. A., & Vannatta, R.A. (2005) *Advanced and multivariate statistical methods: Practical application and interpretation* (3rd Edition). Pyczak.
- Özbay, Ö. (2015). Veri madenciliği kavramı ve eğitimde veri madenciliği uygulamaları. *Uluslararası Eğitim Bilimleri Dergisi, 4(3)*, 262-272.
- Özkan, Y. (2016). *Veri madenciliği yöntemleri* (3. Baskı). Papatya Yayıncılık Eğitim.
- Pal, M., & Mather, P. M. (2003). An assessment of the effectiveness of decision tree methods for land cover classification. *Remote Sensing of Environment, 86(4)*, 554-565.
- Passerini, A., Pontil, M., & Frasconi, P. (2004). New results on error correcting output codes of kernel machines. *IEEE Transactions on Neural Networks and Learning Systems, 15(1)*, 45-54.
- Polat, K., & Güneş, S. (2007). Classification of epileptiform EEG using a hybrid system based on decision tree classifier and fast Fourier transform. *Applied Mathematics and Computation, 187(2)*, 1017-1026.
- Qiu, X. Y., Kang, K., & Zhang, H. X. (2008, June 1-8). *Selection of kernel parameters for KNN* [Paper presentation]. IEEE International Joint Conference on Neural Networks (IJCNN), Hong Kong, China
- Sabzevari, H., Soleymani, M., & Noorbakhsh, E. (2007, July 22-25). *A comparison between statistical and data mining methods for credit scoring in case of limited available data* [Paper presentation]. Proceedings of the 3rd CRC Credit Scoring Conference, Edinburgh, UK.
- Sen, A., Islam, M. M., Murase, K., & Yao, X. (2016). Binarization with boosting and oversampling for multiclass classification. *IEEE Transactions on Cybernetics, 46(5)*, 1078-1091.
- Sharma, A. K., & Sahni, S. (2011). A comparative study of classification algorithms for spam email data analysis. *International Journal on Computer Science and Engineering, 3(5)*, 1890-1895.
- Taruna, S., & Pandey, M. (2014, February 21-22). *An empirical analysis of classification techniques for predicting academic performance* [Paper presentation]. IEEE International Advance Computing Conference, Gurgaon, India.
- Varpa, K., Joutsijoki, H., Iltanen, K., & Juhola, M. (2011). Applying one-vs-one and one-vs-all classifiers in k-nearest neighbour method and support vector machines to an otoneurological multi-class problem. *Studies in Health Technology and Informatics, 169*, 579-583.

- Yurdakul, S. (2015). *Veri madenciliği ile lise öğrenci performanslarının değerlendirilmesi* [Yayımlanmış yüksek lisans tezi, Kırıkkale Üniversitesi]. YÖK Tez Merkezi.
- Zaiane, O. R., & Luo, J. (2001, August 6-8). *Towards evaluating learners' behaviour in a web-based distance learning environment* [Paper presentation]. Proceedings IEEE International Conference on Advanced Learning Technologies (ICALT 2001), Madison, WI, USA,
- Zhang H. (2004, July 25-29). *The optimality of NB* [Paper presentation]. Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference (FLAIRS 2004), Miami Beach, Florida, USA.

Extended Summary

Educational data mining is a discipline that develops methods to obtain information from large amounts of data coming from educational environments and thus tries to better understand students and learning environments (IEDMS, 2016). Decomposing the multi-class problem into two-class classification problems using decomposition methods is one approach used for classification problems in datasets with more than two classes of dependent variable. (Passerini, Pontil & Frasconi, 2004). After the problem is divided into several two-class sub-problems, the results obtained from the classifiers used for each sub-problem are combined to make a multi-class prediction (Lorena et al., 2008).

OVA and OVO are the two most prevalent decomposition techniques in the literature. Given a problem with k classes in the OVA strategy, k binary classifier $f_i(x)$ is generated. Each of these estimators i , is trained to separate the class from the remaining classes. In the OVO strategy, $k \cdot (k-1)/2$ binary classes are created for k classes and this number of binary classifiers is used. Each of these classifiers is used to separate i and j class pairs (Arruti et al., 2014).

The classifiers used in the study are J-48, KNN, LR, NB, and DVM algorithms. Some metrics are used when determining and evaluating the performances of the algorithms used. To evaluate the performance of the algorithms, accuracy, precision, and sensitivity, F-measure and kappa metrics, and MAE, RAE, RMSE, and RRSE, which are metrics that evaluate the error amount of the models, were used in this study. To calculate these values, a complexity matrix is created (Coşkun & Baykal, 2011).

The aim of this study is to compare the performances of both OVO, and OVA strategies used for the classification of multiclass data and some binary classifiers applied under these strategies.

In the study, the performance of some data mining algorithms is compared within the framework of OVA and OVO decomposition strategies to classify according to PISA science achievements. This study used a descriptive research approach. Samples containing missing data that were deleted from the 2015 PISA Turkey data, which included 5895 samples and a data set containing 3459 samples were obtained. Then, 26 independent and 1 dependent variable determined from these data were defined categorically. Five different algorithms determined under the OVA and OVO strategies were applied to the data set. Default parameters are used for all algorithms. Among the test options, 10-fold cross-validation and voting among the combining strategies were used.

The LR algorithm has the highest accuracy value when using the OVA method. The DVM, NB, J-48, and KNN algorithms follows this algorithm, respectively. In terms of MAE and RAE error values, the algorithms are listed as DVM, NB, LR, J-48 and KNN. For RMSE and RRSE error values, the algorithms are listed as LR, NB, J-48, DVM and KNN. In terms of Kappa metric, NB algorithm is followed by LR, J-48, KNN and DVM algorithms, respectively. The KNN algorithm, with a value of 0.3413, has the largest mean error under the OVA strategy. While the LR algorithm, with a value of 0.2793, is the algorithm with the lowest mean error. The algorithms in between are j-48 algorithm in the 2nd place with the value of 0.3014, the SVM algorithm in the 3rd place with the value of 0.29785 and the NB in the 4th place with the value of 0.2858, respectively.

The algorithm with the highest accuracy value of the algorithms under the OVO strategy is the LR algorithm. This algorithm is followed by DVM, NB, J-48 and KNN algorithms, respectively. In terms of the MAE error metric, the LR algorithm is followed by DVM, NB, J-48, and KNN algorithms, respectively. For RAE, RMSE, and RRSE metrics, the algorithms are listed as LR, DVM, J-48, NB, and KNN. According to the Kappa criterion, the ranking is NB, LR, J-48, KNN, and DVM. The algorithm with the highest mean error value under the OVO strategy is the KNN algorithm with a value of 0.34925. While the algorithm with the lowest mean error value is the LR algorithm with a value of 0.3225. Algorithms in between are ranked as NB algorithm in the 2nd place with the value of 0.327, the J-48 algorithm in the 3rd place with the value of 0.32535, and the SVM in the 4th place with the value of 0.3237, respectively.

The most effective algorithm, when both strategies and all algorithms are taken into account in terms of accuracy value, is the LR algorithm under the OVO strategy. Considering the precision values, LR and NB algorithms are the most successful algorithms under the OVA strategy. In terms of sensitivity values, the LR algorithm can be considered as the most successful algorithm under the OVO strategy. In terms of F measure, it can be said that the NB algorithm is the most successful algorithm under the OVA strategy.

Studies comparing the performances of algorithms show a wide variety of results. Since the data set used, the algorithms compared and the selected parameters are different, this diversity is quite normal. However, there are also researchers who criticize these studies because of these differences. Hand (2006) stated that such comparison studies create illusions because many situations such as parameters and preprocessing on the data set are different. In addition, the biased behavior of the newly developed algorithms for more successful performance and the fact that the data sets used in these studies are not real data sets, cause criticism of performance comparison studies.

However, despite all of these concerns, research on algorithm comparison has been recognized in the literature and is still a subject of study. The idea that dataset-successful algorithm pairs can be formed, causes such studies to be produced and it is thought that performance comparison studies will clarify the dataset-successful algorithm matches, over time.