Sınıflandırıcı Toplulukları için Yarı Rastgele Altuzaylar

A Semi-Random Subspace Method for Classification Ensembles

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

***Özetçe*—Topluluk algoritmalarının başarıları iki temel ölçüte dayanır. İlki topluluk içindeki temel sınıflandırıcıların başarıları, ikincisi ise temel öğrenicilerin kararlarının birbirlerinden farklılığıdır. Rastgele Altuzaylar, yüksek farklılık üreterek başarılı olmaktadırlar. Bu çalışmada ise bu farklılığı daha da arttıracak bir yöntem önerilmiş ve 36 sınıflandırma veri kümesi üzerinde orijinal Rastgele Altuzaylarla karşılaştırılmıştır. Denemeler sonucunda önerilen Yarı Rastgele Altuzayların daha başarılı olduğu ancak başarı artışının temel öğrenici sayısıyla ters orantılı olduğu görülmüştür. Bu durum, az sayıda temel öğrenicinin kullanılacağı uygulamalar için Yarı Rastgele Altuzayların iyi bir tercih olduğunu göstermektedir.**

Anahtar Kelimeler—Sınıflandırma Toplulukları, Rastgele Altuzaylar, Karar Ağaçları, Makine Öğrenmesi, Örüntü Tanıma, Yapay Zeka

***Abstract*— The performance of ensemble algorithms is related with two terms: the individual accuracy of base learners and the diversity of their results. Random Subspace algorithm owes its success to the diversity. In this study, we propose a method (Semi Random Subspace) which increases its diversity. We compare our method and original Random Subspace over 36 datasets. The experiments show that our method is superior to the original Random Subspace. But its advantage is limited with the size of the ensemble. In this situation, we can say that Semi Random Subspace is suitable choice for the small ensembles.**

Keywords— Classifier Ensembles, Random Subspace, Decision Trees, Machine Learning, Pattern Recognition, Artificial Intelligence

I. GİRİŞ

Şirketleri tek bir kişi yerine yönetim kurulları yönetir. Demokrasilerde tek bir kişi değil, bir meclis seçilir. Önemli bir karar verecek doktor meslektaşlarıyla konsültasyon yapar. Hayatımızda önemli bir karar vermeden önce birçok kişiye danışırız. Bu ve benzeri örnekler makine öğrenmesi alanındaki topluluk algoritmalarının altyapısını oluştururlar. Topluluk algoritmaları, tek bir öğrenici yerine birçok öğreniciyi eğitip, bunların verdikleri kararları birleştirmektedir [1, 2, 3].

978-1-4673-5563-6/13/$31.00 ©2013 IEEE

Topluluk içindeki öğrenicilerin aynı sonucu üretmeleri durumunda bu aynı kararları birleştirmenin işe yaramayacağı açıktır. Topluluk içindeki öğrenicilerin bir örnek için farklı kararların nasıl üretileceği topluluk algoritmalarının ana sorunlarından biridir. Öğrenicileri farklı eğitim kümeleriyle eğitip böylece farklı modeller öğrenmelerini sağlamak en çok kullanılan yöntemdir. Örneğin Breiman tarafından önerilen [4] Bagging'de, N adet eğitim örneğinden her bir öğrenici için yine N adedi yerine konarak seçilmektedir. Ho tarafından önerilen Rastgele Altuzaylarda [5], öğrenicilerin her biri orijinal özelliklerin rastgele seçilen yarısı ile eğitilmektedir. Rastgele Ormanlarda [5] ise öğreniciler karar ağaçlarıdır. Ağaçların düğümlerinin belirlenmesinde tüm özellikler yerine özelliklerin bir altuzayında arama yapılır. Görüldüğü gibi topluluk algoritmaları genelde özelliklerin ya da örneklerin altkümeleriyle öğrenicilerini eğitmektedir.

Bu çalışmada Rastgele Altuzayların öğrenicilerinin eğitiminde kullanılan özellik altuzaylarının birbirlerinden farklılıklarını arttırmak için bir yöntem önerilmiştir. 2. bölümde yöntemin ayrıntıları verilmiş ve sınırları incelenmiştir. 3. bölümde önerilen algoritmanın üzerinde denendiği veri kümeleri tanıtılmış, 4. bölümde elde edilen sonuçlar verilmiştir. Son bölümde ise elde edilen sonuçlar yorumlanmıştır.

II. RASTGELE ALTUZAYLAR

Rastgele Altuzaylar her bir öğrenicisine orijinal özelliklerin rastgele seçilmiş bir altkümesiyle eğitmektedir. Bu altkümenin boyutunun ne olması gerektiği konusunda Ho yaptığı çeşitli deneyler sonucunda orijinal özellik sayısının yarısının seçilmesinin iyi olduğunu söylemiştir [5]. Buna göre orijinal özellik sayısına d denirse, her bir öğrenici d adet özellikten rastgele seçilmiş d/2 adedi ile eğitilmektedir. Topluluk algoritmalarının başarılı olması için öğrenicilerin kararlarının birbirlerinden farklı olması gerektiğine değinmiştik. Rastgele Altuzaylarda öğrenicilerin kararlarının birbirlerinden farklı olmasını sağlayan mekanizma, seçilen özellik altkümelerinin ortak eleman sayısının azlığıdır. Her bir öğrenici için rastgele

d/2 adet özellik seçildiği için 2 öğrenicinin eğitildiği ortak özellik sayısı ortalama d/4'tür.

Bir topluluğun kararlarının birbirlerinden farklılığı, tüm olası öğrenici çiftlerinin farklı kararlarının ortalaması ile ölçülür [2]. Rastgele Altuzaylarda kararların farklılığı, farklı özellik altuzaylarıyla sağlandığından, böyle bir topluluğun kararlarının farklılığı seçilen altuzayların birbirlerinden farklılıklarıyla ölçülebilir. Örneğin Tablo 1'de 4 adet öğrenicisi olan bir toplulukta her bir öğrenici için seçilen özellik altuzayının diğer tüm altuzaylarla ortalama ortak özellik sayısı verilmiştir.

TABLO I. RASTGELE ALTUZAYLARDA ORTAK ÖZELLIK SAYILARI

	2.altuzay	**3.altuzay**	**4.altuzay**
1.altuzay	d/4	d/4	d/4
2.altuzay		d/4	d/4
3.altuzay			d/4

Tablo 1'de 1.altuzay, 1.öğrenicinin eğitildiği altuzayı göstermektedir. i.altuzayla, j.altuzayın kesişimindeki sayı ise i, altuzayla, j.altuzayın ortalama kaç ortak elemanının olduğunu göstermektedir. örneğin orijinal özellik sayısı 100 ise, her bir öğrenici 50 özellikle eğitilirse, rastgele seçilen 2 altuzayın ortak özellik sayısı 25 olacaktır. Buna göre Tablo 1'deki topluluk için ortalama ortak özellik sayısı d/4'tür. Bu sayı öğrenici sayısına duyarlı değildir. Topluluktaki öğrenici sayısı arttırılsa da ortalama ortak özellik sayısı yine d/4 olacaktır. Yarı Rastgele Altuzaylar ismiyle önerdiğimiz yöntem bu ortak özellik sayısını azaltmayı amaçlamaktadır.

III. YARI RASTGELE ALTUZAYLAR

Altuzaylardaki ortak özellik sayısını azaltmak için denenebilecek bir şey altuzayın boyutunu azaltmaktır. Örneğin orijinal d adet özellikten d/2'si yerine d/4'ü seçilirse ortak özellik sayısı d/8'e düşer. Ancak bu durumda tekil öğrenicilerin her birinin başarısı çok azalacağından topluluğun başarısı artmayacaktır. Yarı Rastgele Altuzaylar, Ho'nun önerdği altuzay boyutunu (d/2) azaltmadan, ortak özellik sayısını azaltmayı amaçlamaktadır. Bunun için topluluğun yarısı rastgele altuzaylarla, diğer yarısı da ilk yarısı tarafından seçilmeyen altuzaylarla oluşturulmaktadır. Örneğin 4 öğreniciden oluşan bir toplulukta orijinal özellik sayısı 8 olsun. Önce 2 adet rastgele özellik altuzayı belirlenir. Ardından her birinin tam tersi seçim yapmış 2 adet daha özellik altuzayı oluşturulur. Bu ters seçimlerle oluşturulmuş altuzaylar, tersleriyle ortak elemanlar içermeyeceklerdir. Örneğin 1. altuzay 1-2-7-8 özellikleriyle, 2. altuzay 1-3-4-8 özellikleriyle oluşturulursa, 3. altuzay 1. altuzayın tam tersi 3-4-5-6'yla, 4.altuzay ise 2.altuzayın tam tersi 2-5-6-7 olacaktır. Durumu genelleştirilmiş hali Tablo 2'de verilmiştir.

TABLO II. YARI RASTGELE ALTUZAYLARDA ORTAK ÖZELLIK SAYILARI

	2.altuzay	**3.altuzay**	**4.altuzay**
1.altuzay	d/4	0	d/4
2.altuzay		d/4	0
3.altuzay			d/4

Tablo 2'de 3. altuzay 1.altuzayın ters seçiminden, 4'te 2'nin ters seçiminden oluşturulduğu için 1 ile 3'ün, 2 ile 4'ün ortak özellikleri yoktur. Ve bu topluluğun ortalama ortak özellik sayısı d/6 (=4*(d/4)/6) olarak hesaplanır. Görüldüğü gibi Yarı Rastgele Altuzay yöntemiyle, altuzay boyutu azaltılmadan ortak özellik sayısı ortalaması d/4'ten d/6'ya düşürülmüştür. Bu azalma, kararların benzerliğini azaltacak ve topluluğun performansını arttıracaktır. Peki bu azalmanın öğrenici sayısıyla bir ilişkisi var mıdır?

Ortak özellik sayılarını gösteren matriste, Yarı Rastgele Altuzaylar kullanıldığında öğrenici sayısının yarısı tane 0 olacaktır. Diğer tüm değerler yine d/4 olacaktır. Altuzay çiftleri sayısı, topluluğun boyutu T ise, T*(T-1)/2'dir. Buna göre ortalama ortak özellik sayısındaki azalma miktarına K denirse, orijinal Rastgele Altuzaydaki ortalama ortak özellik sayısı (RS) ile Yarı Rastgele Altuzaydaki ortalama ortak özellik sayısı (SRS) farkına eşit olacaktır. Eşitlik 1'de bu azalma miktarı gösterilmiştir.

$$K = RS - SRS \tag{1}$$

Eşitlik 1'deki RS ve SRS'nin değerleri Eşitlik 2 ve 3'te verilmiştir.

$$RS = d/4 \tag{2}$$

$$SRS = \frac{(0*T/2) + (\frac{T*(T-1)}{2} - \frac{T}{2})*(d/4)}{\frac{T*(T-1)}{2}} \tag{3}$$

Eşitlik 3'te sadeleştirmeler yapıldığında Eşitlik 4 elde edilir.

$$SRS = \frac{T-2}{T-1}*(d/4) \tag{4}$$

Buna göre K değeri Eşitlik 5 ile hesaplanır.

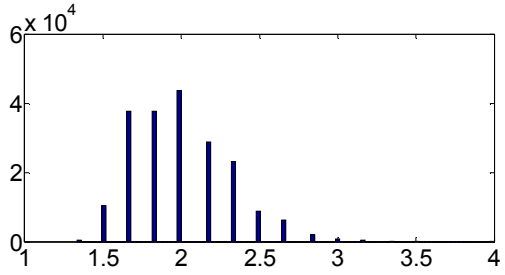
$$K = (d/4) - \frac{T-2}{T-1}*(d/4) = (1 - \frac{T-2}{T-1})*(d/4) = \frac{1}{T-1}*(d/4) \tag{5}$$

Buna göre örneğin T=2 için K (ortak özellik sayısındaki azalma) değeri d/4 olmalıdır. Diğer bir deyişle ortak özellik sayısı 0 (=d/4-d/4) olmalıdır. Gerçektende 2 temel öğrenici olduğunda özelliklerin bir yarısını biri, diğer yarısını diğeri öğreneceğinden ortak özellik sayısı 0 olacaktır.

Eşitlik 5'te bulunan formül incelendiğinde ortak özellik sayısındaki azalmanın T ile ters orantılı olduğu görülmektedir. Buna göre T'nin küçük değerleri için azalma çok olurken, T'nin büyük değerleri için az olacaktır.

"Ortak özellik sayısını önerdiğimiz yöntemden daha fazla azaltabilmek mümkün müdür?" sorusuna cevap aramak için deneysel bir çalışma yapılmıştır. 8 özellikli bir veri kümesi için, her biri rastgele seçilmiş 4 özellikle eğitilen 4 temel öğrenici tasarlanmıştır. Eğitimlerinde kullanılan özellik

altuzaylarının ortalama ortak eleman sayıları bulunmuştur. Bu işlem 200 bin kez tekrarlanmıştır. Bulunan 200 bin ortak eleman sayısının histogramı Şekil 1'de gösterilmiştir.



Şekil 1. Rastgele altuzaylarda ortak özellik sayılarının dağılımı

Şekil 1 incelendiğinde en düşük ortak özellik sayısı 1,33, en yüksek eleman sayısı 4'tür. Ortalaması ise beklediğimiz gibi 2'dir. Önerdiğimiz Yarı Rastgele Altuzay yönteminin bu durum için üreteceği ortak özellik sayısı Eşitlik 4'ten hesaplanırsa 4/3=(8/4)*(4-2)/(4-1) olarak bulunur ki bu da histogramın en düşük değeri olan 1,33'e eşittir. Buradan önerdiğimiz yöntemin olası en az ortak özellik sayısını ürettiği görülmektedir.

IV. KULLANILAN VERİ KÜMELERİ

Algoritmaların performanslarını karşılaştırmak için UCI [7] veri tabanından 36 veri kümesi kullanılmıştır. Bu veri kümelerinin isimleri, örnek, özellik ve sınıf sayıları Tablo 3'de verilmiştir.

TABLO III. DENEMELERDE KULLANILAN 36 VERI KÜMESI

Veri Kümesi	Özellik sayısı	Sınıf sayısı	Örnek sayısı
abalone	11	19	4153
Anneal	63	4	890
audiology	70	5	169
Autos	72	5	202
balance-scale	5	3	625
breast-cancer	39	2	286
breast-w	10	2	699
col10	8	10	2019
Colic	61	2	368
credit-a	43	2	690
credit-g	60	2	1000
d159	33	2	7182
diabetes	9	2	768
Glass	10	5	205
heart-statlog	14	2	270
hepatitis	20	2	155
hypothyroid	32	3	3770
ionosphere	34	2	351
iris	5	3	150
kr-vs-kp	40	2	3196
labor	27	2	57
letter	17	26	20000
lymph	38	2	142
mushroom	113	2	8124
primary-tumor	24	11	302
ringnorm	21	2	7400
segment	19	7	2310
sick	32	2	3772
sonar	61	2	208
soybean	84	18	675
splice	288	3	3190
vehicle	19	4	846
vote	17	2	435
vowel	12	11	990
waveform	41	3	5000
Zoo	17	4	84

V. DENEYSEL SONUÇLAR

Yöntemin etkinliğini ölçmek için Rastgele Altuzay ve Yarı Rastgele Altuzay topluluklarının 6, 10 ve 20 temel öğrenici içeren halleri birbirleriyle karşılaştırılmıştır. Toplamda 6 topluluk algoritması 36 veri kümesi üzerinde çalıştırılmıştır. Denemelerde temel öğrenici olarak bir karar ağacı olan CART [8] kullanılmıştır. Altuzay boyutu orijinal boyutun (d) yarısı seçilmiştir. Bir algoritmanın bir veri kümesi üzerindeki performansı 5*2 çapraz geçerlemeden [9] elde edilen 10 değerin ortalaması olarak belirlenmiştir. Tablo 4'te her bir veri kümesi üzerinde, Yarı Rastgele Altuzayın, Rastgele Altuzayın performansını ne oranda değiştirdiği verilmiştir.

TABLO IV. PERFORMANS DEĞIŞIM ORANLARI (%)

Veri Kümesi	6 temel öğrenici	10 temel öğrenici	20 temel öğrenici
abalone	1,416	-0,426	-0,537
Anneal	0,369	0,183	0,224
audiology	9,417	0,446	-0,993
Autos	1,699	-1,598	-0,276
balance-scale	5,51	2,435	2,859
breast-cancer	2,378	0,984	0,288
breast-w	0,631	0,387	0,24
col10	0,428	0,406	0,591
Colic	0,892	3,029	-0,061
credit-a	1,253	0,903	0,058
credit-g	-0,449	-0,082	-0,054
d159	-0,081	0,223	0,231
diabetes	0,404	-0,183	-0,137
Glass	0,734	2,634	0
heart-statlog	1,261	0,652	1,404
hepatitis	-1,567	2,397	0,467
hypothyroid	0,291	-0,859	-0,083
ionosphere	-0,374	0,629	-0,12
iris	-0,844	0,566	-0,286
kr-vs-kp	1,715	2,888	0,446
labor	0,401	1,615	-0,39
letter	1,411	0,468	0,737
lymph	-1,043	1,379	-0,344

mushroom	0,02	-0,01	-0,01
primary-tumor	1,751	-0,454	0,437
ringnorm	0,659	0,574	0,219
segment	0,564	-0,052	0,657
sick	0,655	-0,649	0,114
sonar	1,462	2,241	0,875
soybean	0,533	0,066	-0,293
splice	0,315	0,372	0,168
vehicle	0,17	-0,138	1,068
vote	0,992	0,679	-0,241
vowel	0,029	1,097	0,024
waveform	0,714	0,726	0,521
Zoo	-1,237	-0,981	-0,731
ortalama	0,902	0,626	0,197

Tablo 4'teki performans değişimleri (r) hesaplanırken Eşitlik 6 kullanılmıştır.

$$r = 100 * (E_{SRS} - E_{RS}) / E_{RS} \quad (6)$$

Eşitlik 6'daki ESRS, Yarı Rastgele Altuzayların 5*2 çapraz geçerleme ile elde edilmiş 10 sınıflandırma başarısının ortalamasını, ERS ise, Rastgele Altuzayların başarı ortalamasını göstermektedir.

Tablo 4'deki sonuçlar incelendiğinde, temel öğrenici sayısının artışıyla, performanstaki artışın azaldığı görülmektedir ki bu Eşitlik 5 in deneysel olarak da doğrulanması anlamına gelmektedir.

Performansları karşılaştırmak için kaç veri kümesinde daha başarılı oldukları ölçütü de kullanılmış ve Tablo 5'te bu değerler verilmiştir.

TABLO V. YARI RASTGELE ALTUZAYLARIN DAHA BAŞARILI/BAŞARISIZ OLDUĞU VERI KÜMESI SAYILARI

	Ortalamaya göre Başarılı/Başarısız	**İstatistiksel anlamlı olarak Başarılı/Başarısız**
6 temel öğrenici	29/7	3/0
10 temel öğrenici	25/11	2/0
20 temel öğrenici	20/16	1/0

Tablo 5'teki değerlere göre 6 temel öğrenici içeren Yarı Rastgele Altuzaylar, Rastgele Altuzayları 36 veri kümesinin 29'unda ortalama başarıya göre geçmiştir ve bunlardan 3'ünde istatistiksel olarak anlamlı farklılık bulunmaktadır. 7 veri kümesinde ise Rastgele Altuzaylar daha başarılıdır ancak bunların hiçbiri istatistiksel olarak anlamlı değildir.

Tablo 5'teki değerler temel öğrenici sayılarına bağlı olarak incelendiğinde yine Eşitlik 5'i doğruladıkları görülmektedir. Temel öğrenici sayısı arttıkça Yarı Rastgele Altuzayların daha başarılı olduğu veri kümesi sayısı azalmaktadır.

VI. SONUÇ

Öğrenme topluluklarında temel öğrenici sayısının artışı topluluğun performansı üzerindeki olumlu etkisi bilinmektedir [2]. Öğrenci sayısının artışı topluluk algoritmalarının performans farklılıkları arasındaki farkları da azaltmaktadır. Banfield ve arkadaşlarının yaptığı çalışmada temel öğrenici sayısını 1000'e kadar arttırdığında topluluk algoritmalarının performansları arasındaki istatistiki olarak anlamlı farklılıkların kaybolduğunu göstermiştir [10]. Ayrıca yüksek performans için temel öğrenici sayısını arttırmak eğitim ve test sürelerini de arttırmaktadır. Bu sebeplerle az sayıda temel öğrenici ile yüksek performansa ulaşan topluluklar aranır olmuştur. Bu çalışmada Rastgele Altuzay algoritmasının alt uzay seçimi için yeni bir versiyonu önerilmiş ve daha başarılı sonuçlar elde edilmiştir.

Önerilen yöntemde altuzay seçimleri orijinal Rastgele Altuzay algoritmasındaki gibi birbirinden bağımsız değil, birbirinin tersi olan ikili seçimlerle yapılmaktadır. Bu sayede altuzayların ortak eleman sayısı ortalaması azaltılmakta ve dolayısıyla temel öğrenicilerin kararlarının birbirlerinden farklılığı arttırılmaktadır. Altuzaylardaki özellik sayısı aynı bırakıldığından temel öğrenicilerin ortalama performansı değişmemektedir. Topluluk algoritmalarının performansını etkileyen bu iki faktörden biri (temel öğrenicilerin ortalama başarısı) aynı bırakılıp, diğeri (temel öğrenicilerin kararlarının farklılığı) arttırıldığından topluluğun performansı da artmaktadır.

Önerdiğimiz yöntemin performans artışının topluluktaki temel öğrenici sayısıyla azaldığı hem teorik olarak hem de deneysel olarak gösterilmiştir.

Sonuç olarak temel öğrenici sayısının az olmasının istendiği durumlarda Yarı Rastgele Altuzayların, Rastgele Altuzaylara göre daha iyi bir tercih olduğu söylenebilir.

Gelecek çalışma olarak, bu yöntemin rastgele seçim içeren diğer topluluk algoritmalarında da kullanılması düşünülmektedir.

KAYNAKÇA

[1] Brown, G., "Ensemble Learning", Encyclopedia of Machine Learning, Springer Press, 2010.

[2] Kuncheva, L., Combining Pattern Classifiers Methods and Algorithms, Wiley-Interscience, 2004.

[3] Haberdar, H. and Shah, S. K., "Video synchronization as one-class learning", 27th Conference on Image and Vision Computing New Zealand (IVCNZ '12) ACM, New York, NY, USA, 469-474, 2012.

[4] Breiman, L., "Bagging predictors", Machine Learning, 24(2), 1996.

[5] Ho, T. K., "The random subspace method for constructing decision forests", IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(8), 1998.

[6] Breiman, L., "Random Forests", Machine Learning, 45(1), 2001.

[7] Blake, C. L., Merz, C. J., UCI repository of machine learning databases, 1998.

[8] Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J., Classification and regression trees. Monterey, CA: Wadsworth & Brooks/Cole Advanced Books & Software, 1984.

[9] Alpaydin, E., "Combined 5 x 2 cv F test for comparing supervised classification learning algorithms", Neural computation,11(8),1999.

[10] Banfield, R.E., Hall, L.O., Bowyer, K.W., Bhadoria, D., Kegelmeyer, W.P., ve Eschrich, S., "A Comparison of Ensemble Creation Techniques", Proc Fifth Int'l Workshop Multiple Classifier Systems (MCS '04), 2004.