

Sürücü Davranışlarının Makine Öğrenmesi Algoritmaları ile Sınıflandırılmasında Pencereleme Yönteminin Etkisi

Asım Sinan YÜKSEL¹, Şerafettin ATMACA²

¹Süleyman Demirel Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 32200, Isparta

²Süleyman Demirel Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 32200, Isparta

(Alınış / Received: 09.05.2019, Kabul / Accepted: 29.07.2019)

Anahtar Kelimeler

Yapay Zekâ,
Makine Öğrenmesi,
Sürücü Davranışları,
Veri Analizi,

Özet: Günlük yaşantımızda ulaşım önemli bir yere sahiptir. Birçok insan kara yolu ulaşımını kullanıp bir noktadan başka bir noktaya kendi aracı ile gitmektedir. Gün içerisinde insanların araç kullanımı sırasında yapmış olduğu bazı riskli hareketler (ani hızlanma, ani yavaşlama, ani sağa dönüş, ani sola dönüş, ani şerit değişimi vb.) sonucunda kazalar yaşanmaktadır. Sürücülerin yapmış olduğu bu riskli hareketler modellenip sürücü profilleri oluşturulabilir. Oluşturulan sürücü profillerine göre sürücüler uyarılabilir ya da araç kullanımı sırasında kazaya neden olabilecek bu davranışları yapmaması için gerekli yaptırımlar uygulanabilir. Bu çalışmada, sürücünün araç kullanımı sırasında sürüş verileri alınmış ve çeşitli makine öğrenmesi algoritmaları ile sürücü davranışları (ani hızlanma, ani sağa dönüş, ani sola dönüş) modellenmiştir. Bu modelleme sonucunda makine öğrenimi algoritmalarının başarı oranları Rastgele Orman için 65,50%, Bayes Ağları için 47.97%, Karar Tabloları için 59.55%, Yapay Sinir Ağları için 55.84% ve Destek Vektör Makineleri için 53.82% olarak bulunmuştur. Sürücü davranışlarının sınıflandırılmasındaki başarıyı artırabilmek için pencereleme yöntemi kullanılmış ve Rastgele Orman'da 89,61%, Bayes Ağları'nda 90.90%, Karar Tabloları'nda 92,20%, Yapay Sinir Ağları'nda 84,41%, Destek Vektör Makineleri'nde 90,90% başarı oranları elde edilmiştir. Sürücü davranışlarının modellenmesinde pencereleme yöntemi olumlu etki oluştururken en yüksek başarı oranına sahip makine öğrenmesi algoritması Karar Tabloları olarak belirlenmiştir.

The Effect of Windowing Method on Classification of Driver Behaviors with Machine Learning Algorithms

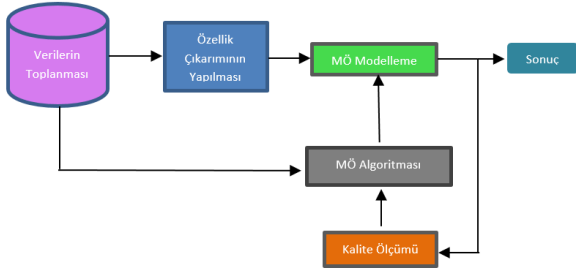
Keywords

Artificial Intelligence,
Machine Learning,
Driver Behavior,
Data Analysis,

Abstract: Transportation has an important place in our daily life. Many people use land transportation and travel from one point to another with their own vehicles. Some risky movements (sudden acceleration, sudden deceleration, sudden right turn, sudden left turn, sudden lane change, etc.) that people did during the day cause accidents. These risky movements of drivers can be modeled and driver profiles can be created. According to the driver profiles generated, drivers may be warned or sanctions may be applied to prevent them from performing these behaviors that may cause accidents. In this study, driver data was obtained during vehicle use and various driver behaviors (sudden acceleration, sudden right turn, sudden left turn) were modeled with machine learning algorithms. As a result, success rates of machine learning algorithms were found as 65,50% for Random Forest, 47.97% for Bayesian Networks, 59.55% for Decision Tables, 55.84% for Artificial Neural Networks and 53.82% for Support Vector Machines. In order to increase the success of the classification, windowing method was applied. Final success rates were 89.61% for Random Forest, 90.90% for Bayesian Networks, 92.20% for Decision Tables, 84.41% for Artificial Neural Networks and 90.90% for Support Vector Machines. While windowing method has positive effect in modeling of driver behaviors, machine learning algorithm with highest success rate is determined as Decision Tables.

1. Giriş

Yapay zekânın bir branşı olan makine öğrenmesi 1986 yılında yapılan çalışmalarla başlamıştır. Makine öğrenmesi, veriden karmaşık örüntünün tespit edilmesi işlemidir ve akılcı karar verme için istatistik ve bilgisayarın hesaplama gücünden yararlanır. Makine öğrenmesi teknikleri sınıflandırma problemlerinde başarılı bir şekilde kullanılmaktadır [1, 2, 3]. Makine öğrenmesinin temel çalışma diyagramı Şekil 1’de verilmiştir.



Şekil 1. Makine Öğrenmesinin Temel Çalışma Diyagramı

Makine öğrenmesinde Yapay Sinir Ağları, Bayes Sınıflandırıcısı, Destek Vektör Makineleri, Karar Ağaçları, Karar Tabloları, Rastgele Orman, Rastgele Ağaç gibi yöntemler kullanılmaktadır. Bu yöntemlerden bazıları sınıflandırma, bazıları kümeleme ve bir kısmı da tahminleme yapabilmektedir. Makine öğrenmesi yöntemleri temelinde denetimli, denetimsiz ve pekiştirmeli öğrenim olarak 3 gruba ayrılmaktadır. Denetimli öğrenimde girdi değerleri ve çıktı değerleri verilerek aradaki ilişkiyi öğrenmesi ve en iyi doğruluğa ulaşması hedeflenir. Denetimsiz öğrenimde ise çıktı değerleri olmaksızın girdi değerlerine ait veriler analiz edilerek girdi değerlerinin ilişkisi bulunmaya çalışılır. Bu ilişkiler sayesinde birbirine yakın değerler gruplandırılır yani kümeleme işlemi yapılmış olur. Yeni girdi verileri hangi küme ile ilişkilendirilirse girdi o kümeyle ait olacaktır. Pekiştirmeli öğrenme yönteminde ise hedeflenen çıktıya ulaşmak için çıktı ile girdinin sonucuna bağlı olarak iyi ya da kötü olarak değerlendirilen kriterler kullanılmaktadır [4].

Sürücü davranışlarının modellenmesi üzerine geçmiş yıllarda çeşitli çalışmalar yapılmıştır. Nidhi Kalra, Gunjan Chugh ve Divya Bansal tarafından yapılan çalışmada [5], sürücü davranışları izlenerek çukurluk, tümsek gibi yol koşullarının analizi yapılmıştır. Sonuç olarak bu çalışma, sürücü güvenliği ve yolların bakımı için sürüş davranışlarını ve yol durumunu analiz etmede yardımcı olmuştur. David Hallac ve arkadaşları tarafından yapılan çalışmada [6] tek bir dönüşte toplanan sensör verilerinden belirli bir sürücü grubunun sürücüsünün kimliğini tahmin edilmesi için bir yöntem önerilmektedir. Yöntemi test etmek için, AUDI AG ve Audi Electronics Venture tarafından toplanan yeni bir veri setini analiz edilmiştir. Sonuç olarak kırsal, kentsel ve otoyol

rampalarında gerçekleşen ve en sık yapılan 12 dönüş hareketi incelenerek sürücü profilinin oluşturulması hedeflenmiştir. Bu çalışmada sürücü davranışlarının modellenmesinde en iyi sonucu verebilecek yöntem ve algoritmalar belirlenmiştir. Elde edilen veriler pencereleme yöntemi ile düzenlenmiş ve Rastgele Orman, Bayes Ağları, Karar Tabloları, Yapay Sinir Ağları, Destek Vektör Makineleri gibi makine öğrenmesi algoritmalarıyla modellenmiştir. Modelleme sonucunda en yüksek doğruluk oranı sahip algoritma Karar Tabloları olarak belirlenmiştir.

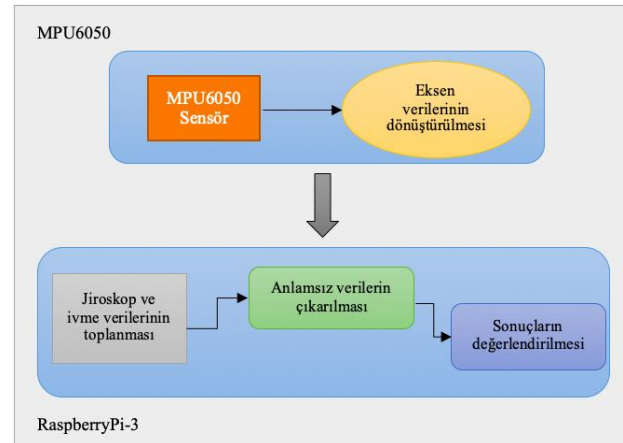
2. Materyal ve Metot

Bu çalışmada sürücünün yol boyunca yapmış olduğu hareketler sensörler aracılığı ile toplanmış ve Weka yazılımı kullanılarak makine öğrenimi algoritmaları ile modellenmiştir [7]. WEKA üzerinde makine öğrenmesi ile ilgili birçok hazır algoritma bulundurmaktadır. WEKA ile veri ön işleme, sınıflandırma, gruplandırma, özellik seçimi veya öznelik çıkarımı gibi işlemler yapılabilmektedir.

2.1. Veri Toplama

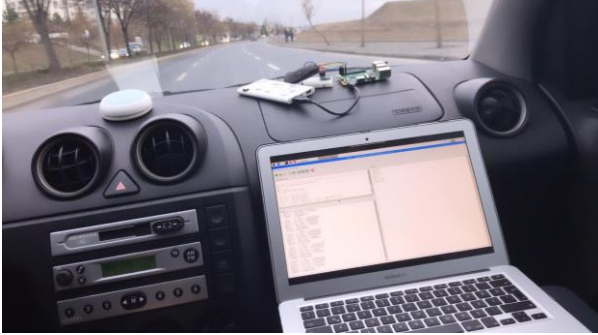
Bu aşamada jiroskop ve ivmeölçer verileri alınmış ve bu verilerden anlamlı verilerin elde edilebilmesi için çalışmalar yapılmıştır. İvmeölçer, üzerine düşen statik (yerçekimi) veya dinamik (aniden hızlanma veya durma) ivmeyi ölçmektedirler. Sensörden alınan değerler m/s² veya yer çekimi (g-Force) türünden ifade edilebilir. Jiroskop ise eksenler etrafındaki dönüşü ve hızı ifade etmektedir. Bu hız dakikadaki dönüş sayısı (RPM) ya da saniyedeki dönüş derecesi (°/sn) olarak ifade edilmektedir.

Jiroskop ve İvmeölçer verilerini elde etmek için Raspberry Pi mini bilgisayarı ve MPU6050 3 eksenli bir jiroskop ve 3 eksenli bir açılabilir ivmeölçer bulunduran 6 eksenli bir sensör kartı kullanılmıştır. Verilerin elde edilmesi için gerekli yapı Şekil 2’de verilmiştir.



Şekil 2. RaspberryPi ve MPU6050 Sensör Devresinin Entegrasyonu

Sürücü davranışlarının belirlenmesinde Hyundai i20 ve Ford Fiesta araçlar kullanılmış ve çeşitli sayılarda ani hızlanma, ani sağa dönüş, ani sola dönüş gibi hareketler 3 kişi tarafından tekrarlanmıştır. Yapılan sürüş verilerine ait jiroskop ve ivmeölçer verileri raspberry pi aracılığıyla toplanmıştır. Şekil 3' de verilerin toplanmasına ait anlık görüntü verilmiştir.



Şekil 3. RaspberryPi ile Anlık Verilerin Toplanması

2.2. Veri Analiz Yöntemleri ve Kullanılan Algoritmalar

Veri analizinde ilk olarak ham verinin tamamı kullanılmış ama istenilen sonuca ulaşamamıştır. Sonuçların daha iyi hale gelmesi için jiroskop x, y, z ve ivmeölçer x, y, z verilerinin ağırlıklı ortalamaları alınmış ve yeni veri seti oluşturulmuştur. Bu veri setinin elde edilmesinde pencereleme yöntemi kullanılmıştır. Verilerin analizi için kullanılan yöntem ve algoritmalar aşağıda verilmiştir.

- Pencereleme Yöntemi (Windowing)
- Yapay Sinir Ağları (Artificial Neural Network)
- Destek Vektör Makineleri (Support Vector Machines)
- Bayes Ağları (Bayesian Network)
- Karar Tabloları (Decision Table)
- Rastgele Orman Algoritması (Random Forest)

2.2.1 Pencereleme Yöntemi

Pencereleme yöntemi küçük bir örneği analiz ederek büyük veri kümelerinin işlenmesini sağlayan örnek seçme ve modelleme yöntemidir. Bu yöntem ile büyük veri kümelerine uygulanan örnek seçme ve modelleme ile olumlu sonuçlar ortaya çıkabilmektedir. Yapılan testler sonucunda Rastgele Orman algoritması ile tüm verinin modellenmesi sonucu doğruluk oranı 65.50% iken aynı algoritmanın pencereleme yöntemi uygulanarak hazırlanmış verilerin modellenmesinin doğruluk oranı 89.61% olmuştur. Pencereleme yöntemi adımları aşağıda belirtildiği şekildedir [8].

- Pencereleme algoritması rastgele örneklerden başlayarak seçilen ilk pencere ile çalışarak bu örneğe dayalı bir model oluşturmaya çalışır.
- Örnek 'Test Veri Seti' üzerinde test edilir.

- Doğru sınıflandırma gerçekleşmişse 'Eğitim Seti'ne eklenir.
- Bu eğitim seti ilk oluşturulan pencere ile birlikte yeni model oluşturmak için kullanılır.
- Yeni model daha sonra test setinin geri kalanında da test edilir ve modelin oluşturulması için gerekli veriler eğitim setine aktarılır.
- Bu işlemler belirli bir doğruluk derecesine ulaşana kadar veya doğruluk arttırılamayacak duruma gelene kadar devam eder.
- Böylece en iyi doğruluk yüzdesi ortaya çıkmış olur.

Pencereleme yönteminde her sürücü davranışı için ayrı pencere boyutları seçilmiştir. Ani hızlanma için dokuz, ani sağa dönüş için on iki, ani sola dönüş için on dört pencere boyutu seçilmiştir. Bunun sebebi her davranışın başlangıç ve bitiş arasında oluşturduğu verilerin farklılığından kaynaklanmaktadır. Tablo 1.' de ani hızlanmaya ait pencerelemenin uygulandığı dokuz satırlık veri seti bulunmaktadır. Bu veri seti 1 seferlik ani hızlanma davranışını ifade etmektedir.

Tablo 1. Ham Verinin Sınıflandırma Tablosu

Sınıf	GyrX	GyrY	GyrZ	İvmX	İvmY	İvmZ
1	-0,92	3,69	0,82	0,16	-0,08	-0,96
1	-0,90	4,53	0,83	0,17	-0,10	-1,01
1	0,78	3,96	0,58	0,32	-0,14	-0,91
1	0,33	4,56	-0,25	0,48	-0,22	-0,93
1	3,35	2,69	-0,10	0,42	-0,25	-0,95
1	-1,50	3,18	-1,65	0,38	-0,14	-0,91
1	1,35	8,72	-0,94	0,40	-0,25	-0,86
1	-1,08	4,10	-0,62	0,34	-0,10	-0,96
1	0,05	5,08	-2,22	0,27	-0,25	-1,00

2.2.2 Yapay Sinir Ağları

Yapay sinir ağları (YSA), insan beyninin ve merkezi sinir sisteminin çalışma prensiplerinden esinlenerek geliştirilmiştir [9]. Biyolojik sistemlerde öğrenme nöronlar arası sinaptik bağlantılar ile sağlanmaktadır. İnsanlar duyu organlarından gelen bilgileri kullanarak sinaptik bağlantılar güncellenir. Öğrenme, örneklerin kullanılması ve sonuçlarla ilişkilendirmesi sonucunda gerçekleşir. Eğitim ise örnekler kullanılarak bağlantı ağırlıklarının belirlenmesi ve en iyi sonuçların elde edilmesine kadar devam eden süreci ifade etmektedir. Yapay sinir Ağları üzerine yapılan ilk çalışmalar nöronların modellenmesi ve bilgisayar sistemlerinde kullanılması ile başlamıştır. [10]. Yapay Sinir Sistemi İşlemci Eleman (Nöron), Toplama Fonksiyonu (Dentrit), Transfer Fonksiyonu (Hücre Gövdesi), Yapay Nöron Çıkışı (Aksonlar) ve Ağırlıklardan (Sinapslar) oluşur. Modelleme ile nöronlar

birbirlerine bağlanır. Her bağlantıya ait nöronun önemini belirten bir sayısal ağırlık vardır. Ağırlıklar, YSA hücresine ait bilginin önemini ve o hücre üzerinde oluşturacağı etkiyi ifade etmektedir [11]. Bu bilgi, öğrenme aşamasında YSA'ya bir eğitim seti sunulduğunda elde edilir. Ağ girdi verisini, çıktı verisi üretilinceye kadar katmandan katmana yayar. Ağırlıkların defalarca belirlenip ayarlanmaları sonucu öğrenim gerçekleşir.

2.2.3. Destek Vektör Makineleri

Destek Vektör Makineleri Cortes ve Vapnik [12] tarafından iki sınıflı gruplandırma için yeni bir makine öğrenmesi yöntemi olarak ilk kez önerilmiştir. Önerilen bu çalışmada makine kavramsal olarak lineer olmayan girdi vektörlerinin çok yüksek boyutta öznelik uzayına eşleştirilmesi ile oluşur. Öznelik uzayı doğrusal karar yüzeyi olarak belirlenir [13]. Karar yüzeyine ait özel nitelikler üst seviyede genelleme yapılmasını sağlar. Destek vektör makinelerinin dayandığı bu yapı eğitim verilerinin en iyi şekilde modellenmesini sağlar. Destek Vektör Makinelerinde, lineer olarak ayrılabilen ya da lineer olarak ayrılamayan verilerin olduğu iki yapı bulunur [14]. Lineer olarak ayrılan verilerin maksimum sınırının bulunması kolaydır. Ancak lineer olarak ayrılmayan verilerin sınırının bulunabilmesi için verilerin lineer olarak ayrılacakları bir uzaya ihtiyaç duyulur.

2.2.4. Bayes Ağları

Ben-Gal'a göre [15], Bayes Ağları olasılıksal grafik modellerin ailesine aittir. Bu grafik yapıları, belirsiz bir alan hakkındaki bilgileri temsil etmek için kullanılır. Özellikle, grafikteki her düğüm rastgele bir değişkeni temsil ederken, düğümler arasındaki kenarlara karşılık gelen rastgele değişkenler arasındaki olasılıksal bağımlılıkları temsil etmektedir. Grafikteki bu tür koşullu bağımlılıklar, istatistik ve hesaplama yöntemleri kullanılarak tahmin edilir. Bayes ağları böylece grafik teorisi, olasılık teorisi ve istatistik ilkelerini birleştirir.

2.2.5. Karar Tabloları

Karar tabloları [16], karar ağaçları [17] veya sinir ağları [18] gibi, tahmin için kullanılan sınıflandırma modelleridir. Karar tabloları, daha yüksek seviye bir tablodaki her bir girişin, başka bir tablo oluşturmak için bir çift ek özneliğin değerleri ile parçalandığı hiyerarşik bir tablodan oluşur. Yapı boyutsal istifleme benzer [19].

2.3.6. Rastgele Orman Algoritması

Denetimli öğrenim algoritmalarından biri olan Rastgele Orman(Random Forest) algoritması karar ağaçlarından birisidir [20]. Ağaç yapısını rastgele seçerek oluşturur ve seçilen rastgele veriden en

iyilerini seçip düğümleri buna göre dallara ayırır. Eğitim verisi asıl veri setinden rastgele seçimler sonucu oluşturulur. Böylece rastgele nitelikler seçilerek ağaçlar geliştirilmiş olur.

3. Bulgular

Veri analizleri için Weka'da yer alan algoritmaların varsayılan parametreleri kullanılmıştır. Raspberry Pi ile elde edilen veriler virgüllü formatta bir dosyaya kaydedilmiştir. Tüm verinin tek seferde analizinden elde edilen doğruluk sonucu yeterli olmadığı için pencereleme yöntemi kullanılarak sınıflandırma başarıları artırılmıştır. Pencere boyutları 5 ve 20 olarak rastgele seçilmiştir. Tablo 2.'de tüm verilerin tek seferde işlenerek çeşitli algoritmalara ait doğruluk oranları verilmiştir. Tablo 3.'de pencereleme boyutu beş seçilmiş ve pencereleme yöntemi uygulanmış modellemenin sonuçları yer almaktadır. Tablo 4.'de ise pencereleme buyutu 20 seçilmiş pencereleme yöntemi uygulanarak oluşan modellemenin sonuçları yer almıştır. Tablo 5.'de ise pencereleme büyüklükleri davranışlara göre(ani hızlanma=9,ani sağa dönüş=14, ani sola dönüş=12) belirlenip pencereleme işlemi uygulanmış ve oluşan verilerin çeşitli algoritmalar kullanılarak sınıflandırmaların doğruluk oranları verilmiştir. Tablolarda hata matrisleri incelendiğinde a=1 ile temsil edilen ani hızlanma, b=2 ile temsil edilen ani sola dönüş ve c=3 ile temsil edilen ani sağa dönüş hareketleridir. Tablo-2'den de görüldüğü üzere pencereleme yöntemi uygulanarak algoritmalarından elde edilen sınıflandırma başarı oranları sırasıyla Rastgele Orman için 89,61%, Bayes Ağları için 90,90%, Karar Tabloları için 92,20%, Yapay Sinir Ağları için 84,41% ve Destek Vektör Makineleri için 90,90% olarak bulunmuştur. Ayrıca sürücü davranışlarının modellenmesinde kullanılan makine öğrenmesi algoritmalarından Karar Tabloları en iyi doğruluk ile sürücünün ani hızlanmasını, ani sağa dönüşünü ve ani sola dönüşünü sınıflandırmıştır.

Tablo 2. Ham Verinin Sınıflandırma Tablosu

Algoritma	Doğru Sınıflandırma (%)	Hata Matrisi
Rastgele Orman	65,50	a b c 144 35 73 a = 1 32 189 67 b = 2 45 55 250 c = 3
Sade Bayes	47,97	a b c 227 10 15 a = 1 180 90 18 b = 2 232 8 110 c = 3
Karar Tablosu	59,55	a b c 78 32 142 a = 1 12 184 92 b = 2 25 56 268 c = 3
Yapay Sinir Ağları	55,84	a b c 117 78 57 a = 1 46 199 43 b = 2

		97 72 181 c = 3
Destek Vektör Makineleri	53.82	a b c 85 13 154 a = 1 13 87 188 b = 2 42 1 307 c = 3

Tablo 3. Pencereleme Boyutu 5 Seçilmiş Sınıflandırma Tablosu

Algoritma	Doğru Sınıflandırma (%)	Hata Matrisi
Rastgele Orman	38.98	a b c 2 1 47 a = 1 2 2 53 b = 2 2 3 65 c = 3
Bayes Ağları	31.63	a b c 47 1 2 a = 1 50 5 2 b = 2 63 3 4 c = 3
Karar Tablosu	39.54	a b c 0 0 50 a = 1 0 0 57 b = 2 0 0 70 c = 3
Yapay Sinir Ağları	21.66	a b c 0 17 0 a = 1 0 13 0 b = 2 0 30 0 c = 3
Destek Vektör Makineleri	40.67	a b c 3 1 46 a = 1 1 5 51 b = 2 3 3 64 c = 3

Tablo 4. Pencereleme Boyutu 20 Seçilmiş Sınıflandırma Tablosu

Algoritma	Doğru Sınıflandırma (%)	Hata Matrisi
Rastgele Orman	39.53	a b c 0 0 12 a = 1 0 0 14 b = 2 0 0 17 c = 3
Bayes Ağları	46.66	a b c 12 0 0 a = 1 14 0 0 b = 2 17 0 0 c = 3
Karar Tablosu	39.53	a b c 0 0 12 a = 1 0 0 14 b = 2 0 0 17 c = 3
Yapay Sinir Ağları	39.53	a b c 0 0 12 a = 1 0 0 14 b = 2 0 0 17 c = 3
Destek Vektör Makineleri	39.53	a b c 0 0 12 a = 1 0 0 14 b = 2 0 0 17 c = 3

Tablo 5. Pencereleme Boyutunun Davranışlara Göre Belirlendiği Sınıflandırılma Tablosu

Algoritma	Doğru Sınıflandırma (%)	Hata Matrisi
Rastgele Orman	89,61	a b c 26 1 1 a = 1 1 23 0 b = 2 4 1 20 c = 3
Bayes Ağları	90.90	a b c 27 0 1 a = 1 1 23 0 b = 2 5 0 20 c = 3
Karar Tablosu	92.20	a b c 28 0 0 a = 1 1 23 0 b = 2 5 0 20 c = 3
Yapay Sinir Ağları	84.41	a b c 15 0 1 a = 1 0 19 0 b = 2 4 0 19 c = 3
Destek Vektör Makineleri	90.90	a b c 27 0 1 a = 1 1 23 0 b = 2 5 0 20 c = 3

4. Tartışma ve Sonuç

Yapılan bu çalışmada sürücünün ani hızlanması, ani sağa dönüşü ve ani sola dönüşü hareketleri çeşitli yöntem ve makine öğrenimi algoritmaları ile modellenmiştir. Pencereleme yöntemi doğru pencereleme boyutları belirlenerek kullanılırsa verilerin modellenmesinde başarılar elde edilebileceği görülmüştür. Elde edilen modelleme sonuçlarına göre sürücülerin sürüş esnasında yapmış olduğu ani hızlanma, ani sağa dönüş, ani sola dönüş gibi agresif hareketler belirlenerek sürücüye ait bir profil oluşturulabilir. Oluşturulan sürücü profiline göre sürücü puanı belirlenebilir, sürücünün sigorta fiyatlandırılması yapılabilir veya sürücü uyarılarak sürücünün kaza yapma riski azaltılabilir. Yapılan testler sonucunda en iyi modelleme ani sağa dönüş hareketinde gerçekleşmiştir.

Gelecek çalışmalarda sürücünün ani şerit değiştirmesi, makas diye tabir edilen hatalı sollama şekli gibi sürücü davranışlarının pencereleme yöntemi ile işlenip çeşitli makine öğrenimi algoritmalarıyla modellenmesi planlanmaktadır.

Teşekkür

5058-YL1-17 numaralı proje ile desteklenen bu çalışmada Süleyman Demirel Üniversitesi Bilimsel Araştırma Projeleri Yönetim Birimi'ne teşekkür ederiz.

Kaynakça

- [1] Junior, J.F., et al. 2017. Driver Behavior Profiling: An Investigation With Different Smartphone Sensors And Machine Learning. PLOS ONE, pp. 1-16.
- [2] Hallac, D., et al. 2016. Driver Identification Using Automobile Sensor Data from a Single Turn. 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 1-6.
- [3] Rizzo, N., et al. 2015. Privacy Preserving Driving Style Recognition. 2015 International Conference on Connected Vehicles and Expo (ICCVE), pp. 232-237.
- [4] Atalay, M., Çelik, E., 2017. Büyük Veri Analizinde Yapay Zekâ Ve Makine Öğrenmesi Uygulamaları, Mehmet Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü Dergisi Cilt.9 Sayı.22 2017 - Aralık (s.155-172)
- [5] KALRA, N., et al. 2014. Analyzing Driving and Road Events via Smartphone. *International Journal of Computer Applications, Volume 98-No.12, pp. 5-9.*
- [6] HALLAC, D., et al. 2016. Driver Identification Using Automobile Sensor Data from a Single Turn. 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 1-6.
- [7] A Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. And Whitten, I.H. 2009. The WEKA Data Mining Software: An Update, SIGKDD Explorations Volume 11, Issue 1, p.10-18.
- [8] Fürnkranz, J., Integrative windowing. *Journal of Artificial Intelligence Research*, 8:129-164, 1998.
- [9] Aydın, D. B., 2016, Yoğun Bakım Ünitesinde İdrar Yolu Enfeksiyonunu Tanımlama için Yapay Sinir Ağı Tabanlı bir Karar Destek Sisteminin Geliştirilmesi, Tez, Bozok Üniversitesi Fen Bilimleri Enstitüsü.
- [10] Freeman, J. A., Skapura, D. M., *Neural Networks Algorithms, Applications and Programming Techniques*. New York, USA: Addison-Wesley Publishing Company, 1991. ACTA INFOLOGICA – 2017 – 1 (2) ISSN: 2602-3563 dergipark.gov.tr/acin 73
- [11] Negnevitsky, M., *Artificial Intelligence: A Guide to Intelligent Systems*. Pearson Education, 2005.
- [12] Cortes, C., Vapnik, V. 1995. Support vector networks, *Machine Learning*, 20, 1-25.
- [13] Cortes, C. and Vapnik, V., 1995. Support Vector Networks, *Machine Learning*, Vol.20, No. 3, pp. 273-297.
- [14] Iranzamini, A. and Lindstrom, J. 2018. Driver Behavior Profiling on Smartphone Data using Machine Learning Methods. *Chalmers University of Technology*, pp.1-68.
- [15] Ben-Gal, I., 2008. *Bayesian Networks*. John Wiley & Sons, Ltd; 2008.
- [16] Kohavi, R., 2005. "The Power of Decision Tables", *Proceedings of the European Conference on Machine Learning, Lecture Notes in Artificial Intelligence 914*, Springer Verlag, Berlin, Heidelberg, NY, pages 174-189.
- [17] Chandrasekar, P., Qian, K., Shahriar, H., & Bhattacharya, P., 2017. Improving the prediction accuracy of decision tree mining with data preprocessing. In 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC) (Vol. 2, pp. 481-484). IEEE.
- [18] Da Silva, I. N., Spatti, D. H., Flauzino, R. A., Liboni, L. H. B., & dos Reis Alves, S. F. 2017. *Artificial neural networks*. Cham: Springer International Publishing.
- [19] Tanha, J., van Someren, M., & Afsarmanesh, H. 2017. Semi-supervised self-training for decision tree classifiers. *International Journal of Machine Learning and Cybernetics*, 8(1), 355-370.
- [20] Genuer, R., Poggi, J. M., Tuleau-Malot, C., & Villa-Vialaneix, N. 2017. Random forests for big data. *Big Data Research*, 9, 28-46.