T.C.
TRAKYA ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ
BİYOİSTATİSTİK ANABİLİM DALI
YÜKSEK LİSANS PROGRAMI

Tez Yöneticisi
Prof. Dr. Necdet SÜT

KARAR AĞAÇLARI İLE LOJİSTİK REGRESYON ANALİZİNİN PERFORMANSLARININ SİMÜLASYON ÇALIŞMASI İLE KARŞILAŞTIRILMASI

(Yüksek Lisans Tezi)

Mehmet KARADAĞ

EDİRNE – 2014

T.C.
TRAKYA ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ
BİYOİSTATİSTİK ANABİLİM DALI
YÜKSEK LİSANS PROGRAMI

Tez Yöneticisi
Prof. Dr. Necdet SÜT

KARAR AĞAÇLARI İLE LOJİSTİK REGRESYON ANALİZİNİN PERFORMANSLARININ SİMÜLASYON ÇALIŞMASI İLE KARŞILAŞTIRILMASI

(Yüksek Lisans Tezi)

Mehmet KARADAĞ

Destekleyen Kurum: TÜBAP

Proje No: 2012/125

EDİRNE – 2014

TEŞEKKÜR

Yüksek lisans tez çalışmamda gerek yazım gerek analiz aşamasında desteğini ve yardımını esirgemeyen akademik danışmanım Sayın Prof. Dr. Necdet SÜT'e, Arş. Gör. Selçuk Korkmaz'a, desteklerinden dolayı TÜBAP'a, bana her konuda sabırla yardımcı olan eşim Arzu Karadağ'a ve aileme teşekkür ederim.

Mehmet KARADAĞ

İÇİNDEKİLER

SİMGE VE KISALTMALAR

AUC : ROC eğrisi altında kalan alan

CART : Classification and Regression Tree (Sınıflandırma ve Regresyon Ağaçları)

CHAID : Chi-squared Automatic Interaction Detection (Otomatik Ki-Kare etkileşim belirleme)

C4.5 : C4.5 Karar Ağacı

C5.0 : C5.0 Karar Ağacı

C(i / j) : Kayıp sınıflandırma değeri bir sınıfın j gözlemi gibi bir sınıfın i gözlemi C(i / j) = 0

$\boldsymbol{f_n}$: n-nci gözlemin frekansının ağırlığı

$\boldsymbol{\hbar}$**(t)** : t-nci düğüme düşen örneklem bilgisi

$\hbar = \{\boldsymbol{X_m}, \boldsymbol{Y_n}\}_{\boldsymbol{n=1}}^{\boldsymbol{N}}$: Bütün örneklem bilgisi

ID3 : ID3 Karar Ağacı

J48 : C4.5 Karar Ağacının Java uygulaması

LR : Lojistik Regresyon

MARS : Multivariate Adaptive Regression Splines (Multivariate Adaptive Regression Splines)

NKD : Negatif kestirim değeri

PKD : Pozitif kestirim değeri

$\boldsymbol{P(j,t), j = \overline{1,j}}$: j-nci sınıf ve t-nci sınıf düğümdeki bir gözlemin olasılığı

$\boldsymbol{P(t)}$: t-nci düğüme düşen gözlem değerinin olasılığı

P(j / t),j= $\boldsymbol{\overline{1,j}}$: t-nci düğümdeki j sınıfındaki gözlemim olasılığı

ROC	: Receiver Operating Characteristic Curve
QUEST	: Quick, Unbiased, Efficient Statistical Tree (Hızlı, Tarafsız, Verimli İstatistiksel Ağaç)
$\boldsymbol{X_m, m = \overline{1, m}}$	:Tahmin edici değişken Bu değişken sıralı ölçülü kategorik sürekli olabilir
$\boldsymbol{W_n}$	: Gözlem ağırlığı, n-nci gözlemle alakalı ağırlık
$\boldsymbol{Y}$	: Bağımlı değişken
$\pi_{(j)}, \boldsymbol{j = \overline{1, j}}$	: Y'nin önceki olasılığı

GİRİŞ VE AMAÇ

Veri madenciliği çeşitli şekillerde elde edilmiş veriyi analiz ederek anlaşılır bir yapıya dönüştürmeyi hedeflemektedir (1). Özellikle tıp ve biyoloji alanında yapılan çalışmalarda, veri setleri oldukça karmaşık bir yapı teşkil etmektedir (2). Bu noktada veri madenciliği sağlık ve tıp alanındaki büyük veri tabanlarından faydalı bilgileri ortaya çıkararak hem tıp hem de hizmet kalitesinin artması bakımından büyük katkılar sağlamaktadır (2).

Verinin içerdiği ortak özelliklere göre ayrıştırılması işlemi sınıflandırma olarak adlandırılır; veri madenciliğinde önemli bir konudur (3). Verilerin sınıflandırılmasında Diskriminant Analizi, Kümeleme Analizi, Faktör Analizi, Uyum Analizi gibi yöntemler kullanılır. Bu yöntemlerin yanı sıra karar ağaçları da verilerin sınıflandırılmasında kullanılan yöntemlerden birisidir (3).

Verilerin içerdiği ortak özellikleri kullanarak söz konusu verileri sınıflandırmak mümkündür (3). Sınıflandırma bir öğrenme algoritmasına dayanır (3). Tüm veriler kullanılarak eğitme işi yapılmaz (3). Bu veri topluluğuna ait bir örmek veri üzerinde gerçekleştirilir (3). Öğrenmenin amacı bir sınıflandırma modelinin yaratılmasıdır (3). Bir başka deyişle sınıflandırma, hangi sınıfa ait olduğu bilinmeyen bir kayıt için bir sınıf belirleme sürecidir (3).

Veri madenciliği Biyotıp, Gen fonksiyonları ve DNA sıralama desenlerinin veri analizlerinde, hastalık tanısında, telekomünikasyon endüstrisinde, finans analizi, astronomi ve birçok alanda uygulanmaktadır (4).

Verileri sınıflandırma yöntemlerinden biri “karar ağaçları” (decision trees) ile sınıflandırma adını taşımaktadır (3). Karar ağacının oluşturulmasında CHAID (Chi-squared

Automatic Interaction Detection), CART (Clasification and Regression Trees), ID3, QUEST (Quick, Unbiased, Efficient Statistical Trees), C4.5, C5.0, gibi algoritmalar kullanılır (5). Bu algoritmaların bazıları aynı zamanda regresyon için de uyarlanabilir (5). Çeşitli algoritmaların ortaya çıkış sebebi, karar ağacı oluşturulurken herhangi bir kökten itibaren ayrışmanın ve dallanmanın hangi kritere göre yapılacağı sorununa farklı yaklaşımlarda bulunmasından kaynaklanmaktadır (5). Karar ağaçlarına ek olarak LR (Lojistik Regresyon) analizi, temelde regresyon analizi olmakla birlikte bir ayırıcı analiz tekniği olma özelliğini de taşımaktadır (6).

Çalışmamızın amacı karar ağacı yöntemlerinden olan CART, CHAID ve C4.5 (java uygulaması J48) ile LR analizinin performanslarını simülasyon verileri kullanarak karşılaştırmaktır. Simülasyon çalışması sonucu ilgili yöntemlerin performansları sensitivity (duyarlılık), specificity (özgünlük) ve ROC eğrisi altında kalan alan yardımıyla karşılaştırılacaktır.

GENEL BİLGİLER

Bilgisayarın yaşamımıza daha çok girmesiyle birlikte pek çok alanda yapılan işlemler sayısal ortamda kayıt altına alınmaya başlanmıştır (7). Bu verileri faydalı bilgiye çevirme ihtiyacı ve edinilen verilerle sahip olunan bilgi arasındaki açığı kapatmak üzere geliştirilen yöntem ve teknikler, veri tabanındaki bilgi keşfi sürecinin konusunu oluşturmuştur (8). Bu süreç içerisinde yer alan veri madenciliği, veri yığınları içinde tek başına bulunmayacak ilişkileri, örüntüleri yani olay dizilerini ve anomalileri keşfetmeyi sağlayan önemli bir tekniktir (8).

Veri madenciliği bilgi teknolojilerinin doğal evriminin bir sonucu olarak da nitelendirilebilir (9). Veri tabanı sistemleri evrimsel yolu izleyerek veri toplama, veri tabanı oluşturma, veri yöntemi (veri saklama ve geri erişim dâhil) ve yüksek veri analizi aşamalarından geçerek günümüze gelmiştir (9). 1960'lı yıllardan itibaren veri tabanı ve bilgi teknolojileri basit dosya işlemlerinden gelişmiş ve güçlü veri tabanı yapılarına doğru gelişim göstermiştir (9). 1970'li yıllarda başlayan veri tabanı sistemlerindeki araştırma ve geliştirme çalışmaları hiyerarşik ve ağ veri tabanı yapılarından ilişkisel veri tabanı, veri modelleme araçlar ve indeksleme yapısına geçişi sağlamıştır (9).

1980'li yılların ortasından itibaren ilişkisel teknolojilerle birlikte yeni ve güçlü veri tabanı sistemleri üzerinde durulmuştur (5). İleri veri modellerindeki araştırmalarla nesneye yönelik, nesne-ilişkisel ve tümdengelim yöntemlerinde gelişmeler sağlanmıştır (5).

Veri madenciliğinin günümüzde yaygın bir kullanım alanı bulunmaktadır (3). Veri madenciliği yardımıyla DNA sıra (veri) analizi yapılabilmektedir. İnsanda yaklaşık 100.000 gen vardır (29). Hastalıklara yol açan gen sıralama örneklerine binlerce gen arasından

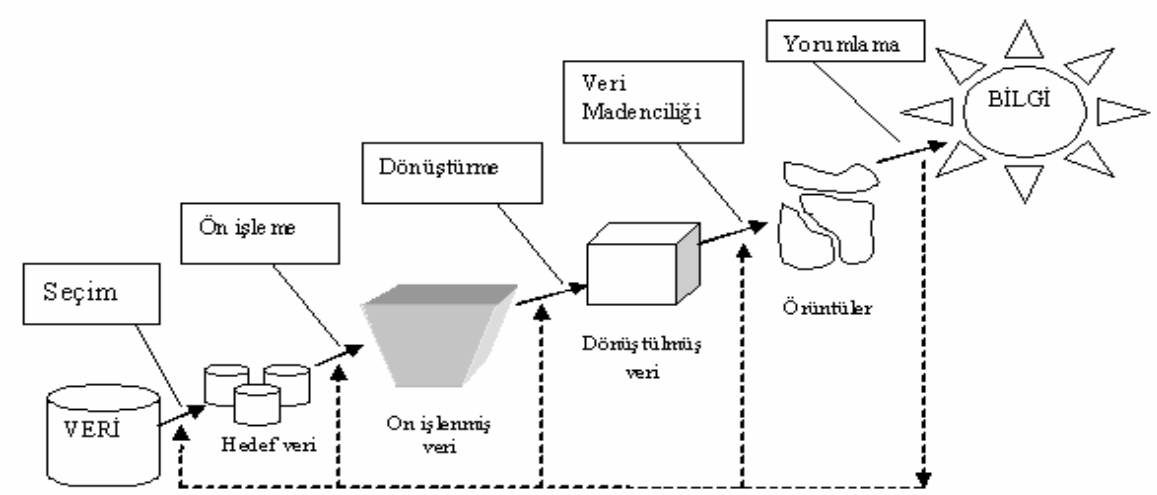
bulmak, tanımlamak oldukça zor bir iştir (29). Veri madenciliği ile geliştirilen sıralama örnek analizi ve benzerlik arama yöntemleri DNA verisi üzerinde analiz yapmayı kolaylaştırır (29).

Veri madenciliği, elektronik hasta dosyalarının oluşturulması hastanın hikâyesine yönelik tüm kayıtların; teşhis tedavi süreçlerinin; laboratuvar sonuçlarının; röntgen, MR gibi görüntü dosyalarının bir tek kayıt içerisinde zamana endeksli olarak hazırlanması verilerin değerlendirilebilmesinde ve hizmet sunumunda büyük önem taşımaktadır (29). Günümüzde bilgi sistemleri ve iletişim teknolojilerindeki gelişmeler sayesinde tıp ve sağlık alanındaki birçok veri sayısal ortamda saklanabilmekte ve kolaylıkla erişilebilmektedir (11).

Bazı hastalıkların %100 kesin teşhisi mümkün olmamaktadır (7). Örneğin gebelik esnasında çocukta oluşabilecek herhangi bir down sendromu riskinin kesin tanısı dış bulgularla sağlanamamaktadır (7). Buradaki dış bulgulardan kasıt, anneden alınacak kan örneği, ultrason ile bebeğin görüntülenmesi, anne adayının yaşı, hamilelik ayı aldığı kilo vs. gibi bulgulardır (7). Ancak bu bulguların hemen hiç biri hekime %100 tanı koyma olanağı vermez; %100 veya %100'e çok yakın bir tanı için anne karnından alınacak sıvının incelenmesi de gerekmektedir (7). Oysa bu işlemde de 1/300 oranında bir düşük riski vardır (7). Dolayısıyla bu işleme girmeden önce hekimin anne karnındaki bebekte down sendromu olduğundan kuşkulanması gerekmektedir (7). Bu aşamada yukarıdaki sözü edilen dış bulgular ve veri madenciliği teknikleri devreye girmektedir (7).

Tıp alanında bunun gibi ameliyat riski taşıyan ancak, ameliyat öncesinde gerçekten ameliyat olması gerektiği tam olarak anlaşılmayan hasta ve hastalık için de veri madenciliği yöntemi kullanılır (7).

Veri tabanlarında bilgi keşfi süreci, veri tabanlarını kullanarak veri tabanlarında istenilen seçim, ön işleme, alt örnekleme, dönüşüm, örüntülerin açığa çıkarılması için veri madenciliği yöntemlerinin (algoritmalarının) uygulanması ve açığa çıkarılan örüntülerin tanımlanması için Veri Madenciliği ürünlerinin yorumlanmasını ihtiva eder (28). Veri tabanlarında bilgi keşfi sürecinin, veri madenciliği bileşeni, veriden hangi örüntülerin aktarılıp, dikkate alınacağının algoritmik anlamda ifadesi olarak değerlendirilmelidir (28). Veri tabanlarında bilgi keşfi sürecinin bütünü, (Şekil 1)'de de görüldüğü gibi, değerlendirme ve madenlenmiş örüntülerin hangilerinin yeni bilgi olarak değerlendirileceğinin olası yorumunu da içerir (28) .



Şekil 1. Veri tabanlarında bilgi keşfi

SINIFLANDIRMADA KULLANILAN KARAR AĞAÇLARI YÖNTEMLERİ

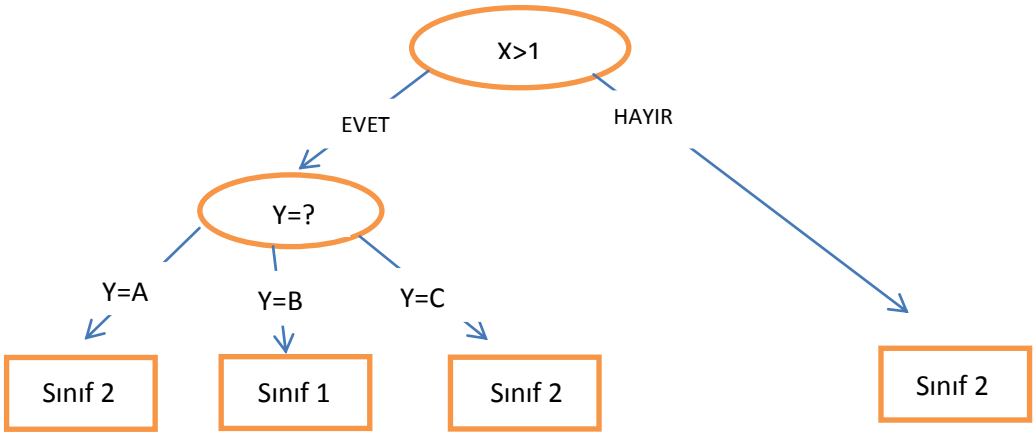
Karar Ağacı (KA), bağımlı değişken üzerindeki farklılıkların maksimize edilmesi amacıyla veri setinin sıralı bir şekilde bölünmesini ifade eder (8). Verileri belli değişken değerlerine göre sınıflandırmaya yarayan karar ağacında kullanılan algoritmalarda girdiler ve çıktılar verilerin belirlenen değişkenleridir ve karar ağacı algoritması çıktı veri değişkenleri için girdi veri değişkenlerini veri yapıları ile keşfeder (8).

Helberg'e (1998) göre karar ağacı karakteristikler kümesi ile kategorik çıktılar arasında bir ilişki bulur (8).

Karar ağacı, veriden sınıflandırıcılar üretmek için kullanılan etkili yöntemlerden biridir (8). Karar ağacı sunumu, en yaygın kullanılan mantık yöntemidir (8). Esas olarak makine öğreniminde ve uygulamalı istatistik literatüründe tanımlanan çok fazla sayıda karar ağacı tümevarım algoritması vardır (8). Bu algoritmalar, bir seri girdi-çıktı kümesinden karar ağacı oluşturan denetlenmiş öğrenme yöntemleridir (8). Tipik bir karar ağacı öğrenme sistemi, araştırma alanının bir kısmında çözüm arayan, yukarıdan aşağıya yöntemini benimser (8). Bu yöntem basit bir ağacın (en basiti olması şart değil) bulunabileceğini garanti eder (8). Bir karar ağacı, değişkenlerin test edildiği yerlerde düğümler içerir (8). Bir düğümden dışa açılan dallar, düğümdeki testin bütün olası sonuçlarına karşılık verir (8).

Karar ağaçları akış şemalarına benzeyen yapılardır (7). Her bir nitelik bir düğüm tarafından temsil edilir (7). Dallar ve yapraklar ağaç yapısının elemanlarıdır (7). En son yapı "yaprak", en üst yapı "kök" ve bunların arasında kalan yapılar ise "dal" olarak isimlendirilir (7). (Şekil 2) üzerinde tipik bir karar ağacı yer almaktadır (3). Karar ağaçları sınıflama algoritmalarını uygulayabilmek için uygun bir alt yapı sağlamaktadır (3). X ve Y'den oluşan

iki giriş niteliğine sahip bir örnek sınıfının basit karar ağacı (Şekil 2) üzerinde görülmektedir (3). X>1 ve Y=B değerini taşıyan örnekler Sınıf 1'de; Y=A ve Y=C koşullarına uygun olanlar Sınıf 2'de yer almaktadır (3). Ancak Y'nin değerini göz önüne almadan X≤1 koşuluna uygun örnekler Sınıf 1'de yer alır (3).



Şekil 2. X ve Y nitelikleri üzerine uygulanan testleri içeren basit bir karar ağacı (3).

Karar ağaçlarının dallanma kriterleri:

Karar ağaçlarında en önemli sorunlardan birisi herhangi bir kökten itibaren bölümlemenin veya bir başka deyişle dallamanın hangi kıstasa göre yapılacağıdır (3). Aslında her farklı kriter için bir karar ağacı algoritması karşılık gelmektedir (3). Söz konusu algoritmaları şu şekilde gruplayabiliriz (3):

a. Entropiye dayalı algoritmalar
b. Sınıflandırma ve regresyon ağaçları (CART)
c. Bellek tabanlı sınıflandırma algoritmaları

Entropiye dayalı bölümlemeyi kullanan algoritmalara örnek olarak ID3 ve onun gelişmiş biçimi olan C4.5 algoritmaları verilebilir (3). Sınıflandırma ve regresyon ağaçları konusunda ise Twoing ve Gini algoritmalarından söz edilebilir (3). Bellek tabanlı sınıflandırma yöntemleri arasında K en yakın komşu algoritması sayılabilir (3).

Karar ağaçları algoritmaları bir sınıflandırma modeline ihtiyaç duyan tahmin görevlerinde kullanılır (10). Sorunların en iyi şekilde çözülebilmesi için durumlar farklı gruplara bölünecek şekilde tasarlanmıştır (10).

Bazı durumlarda modellerin nasıl çalıştığı ile pek ilgilenilmezken, önemli olan bir sınıflandırmanın veya tahminin isabetliliği olabilmektedir (10). Prensip olarak, verilen bir dizi değişkenden oluşturulabilecek pek çok karar ağacı bulunabilmektedir (10). Karar

ağaçlarından bazıları diğerlerine göre daha doğru olurken, en uygun ağacın bulunması, araştırma alanının gittikçe büyüyen boyutu nedeniyle hesaplanamamaktadır (10). Yine de, makul bir süre içinde, makul ölçüde doğru karar ağacının indüklenmesi amacıyla etkin algoritmalar geliştirilmiştir (10). Bu algoritmalar, çoğunlukla, verilerin bölümlere ayrılması için hangi niteliğin kullanılacağına ilişkin olarak en uygun kararlar dizisinin alınmasıyla bir karar ağacı oluşturan strateji yürütmektedir (10). Bu algoritmalardan, İD3, C4.5, J48, CHAID ve CART dâhil olmak üzere, mevcut pek çok karar ağacı endüksiyon algoritmasının temeli olan Hunt'un algoritmasıdır (10). Karar ağacı üreten algoritmaların en iyi bilinenleri ID3 ve C4.5'tir (30).

Karar ağaç endüksiyonu, sınıflandırma modellerinin oluşturulmasına yönelik parametrik olmayan bir yaklaşımdır (26).

Uygun bir karar ağacın bulunması, parametrik olmayan eksiksiz bir sorundur (26). Pek çok karar ağacı algoritmasında, geniş hipotez alanlarındaki arayışları yönlendirmek için buluşsal yöntemlere dayalı bir yaklaşımdan yararlanılmaktadır (26).

Gereğinden fazla değişkenlerin varlığı, karar ağacının doğruluğunu olumsuz yönde etkilemektedir (26).

ID3 Algoritması

Günümüzde ID3 hem akademik, hem de sanayi alanında pek çok sorunu çözme amaçlı kullanılmış, değiştirilmiş, geliştirilmiş ve zaman içerisinde yaygın kullanım alanı bulmuştur (13). ID3 algoritması, ağacın kök düğümündeki düzeltme örnekleriyle başlar (14). Bu örnekleri bölümlemek için bir değişken seçilir (14). Her bir değişken değeri için bir dal oluşturulur ve dal tarafından kendisine yeni bir özellik kazandırılmış olan örnek alt kümeler de yeni oluşturulan alt düğüme yerleştirilir (14). Bir düğümdeki bütün örnekler tek bir sınıfa ait olana kadar algoritma her bir alt düğüme tekrar tekrar uygulanır (14). Karar ağacı yaprağındaki her yol, bir sınıflandırma kuralını ortaya koyar (14). Böyle bir tepeden aşağı karar ağacı çıkarım algoritmasında önemli bir husus, düğümdeki niteliğin seçimidir (14). ID3 ve C4.5 algoritmalarındaki değişken seçimi bir düğümdeki örneklere uygulanan entropi ölçüt bilgisini en aza indirme temeline dayanır (14).

Ağaç yapısında yaprağı olmayan bir düğümde örnekler dallara bölünür ve her bir alt düğüm örneklerin karşılığı olan alt kümeleri elde eder (31). Tek değişkenli bölümlerde kullanılan karar ağaçların basit bir sunum şekli vardır (31). Bu da kullanıcı için, ifade edilen modelin daha iyi anlaşılmasını kolaylaştırır (31). Aynı zamanda modelin ifade etme

yetersizliğine de bir sınırlama getirir (31). Genellikle, özel bir ağaç sunumundaki herhangi bir sınırlama, önemli ölçüde işlev şeklini ve dolayısıyla da modelin yaklaşım gücünü de sınırlayabilir (31). Tek değişkenli bölümleri temel alan karar ağacı oluşturmak için kullanılan ağaç geliştirme algoritmalarından en iyi bilinenlerinden biri de, Quinlan'ın ID3 ve daha gelişmiş şekli C4.5'tir (31). Karar ağacı yapılarında büyüme ve budamayı araştıran yöntemler tipik olarak, mümkün olan modellerin hızla büyüyen alanlarını keşfetmek için de bu algoritmalarda kullanılır (31).

C4.5 algoritması özellikle entropi ölçüsüne göre kural üretmelerinden dolayı daha iyi sonuçlar ürettikleri görülmektedir (17).

Karar ağacında entropi:

Bir sistemdeki belirsizliğin ölçüsüne 'entropi' denir (27). Entropi beklentisizliğin maksimumlaşmasıdır (25). S bir kaynak olsun (27). Bu kaynağın $\{m_1, m_2 \ldots m_n\}$ olmak üzere n tane mesaj üretebildiği varsayılsın (27). Tüm mesajlar birbirinden bağımsız olarak üretilir ve m_i mesajların üretilme olasılıkları p_i'dir (27). P=$\{p_1, p_2 \ldots p_n\}$ olasılık dağılımına sahip mesajları üreten S kaynağının entropisi H (S) aşağıdaki şekildedir (27):

$$H(S) = -\sum_{i=0}^{n} p_i \log_2(p_i)$$

C4.5 Algoritması

C4.5, ID3'ün geliştirilmiş halidir (19). C4.5 eksik ve sürekli değişken değerlerini ele alabilmekte, karar ağacının budanması ve kural çıkarımı gibi işlemleri yapabilmektedir (19). C4.5 algoritması, ID3 algoritmasının bir uzantısıdır ve budama metodolojisi ile sayısal nitelikleri, kayıp değerleri ve gürültülü verileri işlemeyi kapsayan "böl ve yönet" yaklaşımını içermektedir (39). Bölme düğüm stratejisi, bilgi kazanım oranını hesaplamaya dayanmaktadır (39). Temel fikir, kök düğümden bu düğüme olan yolda henüz dikkate alınmamış nitelikler arasında, her bir düğümün en bilgilendirici ilgili nitelik ile ilgili bir soruyu tutmasıdır (39).Yakın zamanda kural türetme hızının ve kalitesinin kendinden önceki versiyonu olan C4.5'ten daha iyi seviyede olan C5.0 ve J48 geliştirilmiştir (19). C5.0 bunlara ek olarak çoklu karar ağaçları tek bir sınıflandırıcı bünyesinde birleştiren destekleme (boosting) adı verilen tekniği de uygulamaya koymuştur (19). Destekleme, farklı sınıflandırıcıları birlikte kullanma yaklaşımıdır (19). Destekleme normalde belirli bir sınıflandırıcıyı çalıştırmak için daha fazla

zaman harcarken doğruluk oranını arttırmaktadır (19). Bazı veri kümelerinde hata oranının, C4.5 ile bulunanın yarısından daha az olduğu görülmüştür (19). Eğitim verisi çok gürültü içerdiğinde destekleme her zaman etkili olmaz (19). Desteklemenin çalışma prensibi, bir eğitim kümesinden birden fazla eğitim kümesinin oluşturulmasıdır (19). Eğitim kümesindeki her kaleme ağırlık tayin edilir (19). Ağırlık, söz konusu kalemin sınıflandırma açısından önemini temsil eder (19). Kullanılan her ağırlıklar kombinasyonu için sınıflandırıcı oluşturulur (19). Böylece aslında çok sayıda sınıflandırıcı oluşturulmuş olur (19). C5.0 ile sınıflandırma yapıldığında her sınıflandırıcıya oy tayin edilir, oylama yapılır ve hedef değişkenler grubu, en çok oy alan sınıfa tahsis edilir (19).

C4.5 algoritması ID3 algoritmasına şu konular açısından üstünlük sağlamıştır (16): Karar ağacı oluştururken kayıp veriler hesaba katılmaz (16). Yani, kazanım oranı hesaplanırken, sadece verileri eksik olmayan diğer kayıtlar kullanılır (16). C4.5 algoritması, kayıp verileri diğer veri ve değişkenler yardımıyla öngörerek kazanım oranı hesaplanmasında kullanılır (16).

Ağacın büyüme işlemi gerçekleştikten sonra hata tabanlı budama işlemi başlar (43). C4.5 sayısal öznitelikleri (değişkenleri) işleyebilir (43). Düzeltilmiş kazanç oranı ölçütü kullanarak eksik değerler içeren bir eğitme kümesinden indükleyebilir (43).

Aşırı uyum (overfitting*)* sebebiyle oluşan hata, C4.5 tarafından geliştirilen yöntemle telafi edilmeye çalışılmaktadır (15).

Alt ağaçları bölme:

T veri kümesine bir X testi uygulandığında $O_1,O_2,...,O_n$ çıktıları elde edilmektedir (24). Bilinmeyen verilerle çalıştığında bu çıktılar sonuç vermez (24). T veri yığınından bilinen bir O_i çıkışı T_i alt kümesini oluşturur (24). O_i çıkışını oluşturan olayların T_i kümesine ait olma olasılığı 1 iken, diğer bütün alt kümelere ait olma olasılığı 0'dır (24).

Bilinmeyen verilerde, bilinmeyen verinin bulunduğu satır ve bu verinin her bir T_i alt ağacında bulunma olasılığı hesaplanır (24). Satır her alt ağaca gönderilir (24). Bu noktada o satırın aynı alt ağaçta olma olasılığı da eklenen satıra işlenir (24). Böylelikle bu satırın olma olasılığı her bir alt kümede 1'den küçük olacak ve her bir alt ağaç için oluşan olasılıklar toplamı 1'e eşit olacaktır (24). Bir satırda birden fazla bilinmeyen değer varsa bu olasılıklar çarpılacaktır (24). Eğer bir satırın ağırlığını w ile gösterirsek, bir sonraki testten oluşacak ağırlık şu şekilde bulunur (24): wi=w x Oi çıkış olasılığı (24).

C4.5 algoritmasında budama:

Basit bir veri yığınından çok büyük bir ağacın elde edilmesine aşırı uyum ya da şişme (overfitting) denir (24). Aşırı uyum, her veri yığını için karşılaşılması mümkün bir sorundur (24). Aşırı uyum, veri yığınındaki gürültüden kaynaklanabileceği gibi seçilen veri kümesinin o olayı temsil edebilme yeteneğinin olmamasından da kaynaklanabilir (24). Aşırı uyum ya da farklı sorunların bir sonucu olarak karar ağacının çok büyük çıkması, anlamsız sonuçlar oluşturabilir (24). Ayrıca diğer önemli bir konu farklı veri kaynaklarından gelen özellik değerleri ölçekleme, birim sistemi veya gösterimdeki farklılıklar yüzünden birbirlerinden farklı olabilirler (44). Örneğin ağırlık özelliği farklı kaynaklarda farklı birim sistemiyle depolanmış olabilir (44). Veri bütünleştirme işlemlerinde verinin bu tür heterojenliği dikkate alınmalıdır (44). Veri dönüştürme; düzeltme, birleştirme, genelleştirme ve normalleştirme gibi işlemlerin bir veya bir kaçını içerir (45).

Karar ağacı budanmasında yapılması gereken asıl görev, bir ya da fazla alt ağacı çıkarıp bunları yapraklarla değiştirerek karar ağacını sadeleştirmektedir (24). Alt ağacın bir yaprakla değiştirilmesinde algoritmanın, öngörülen hata oranını düşürmesi ve sınıflandırma modelinin kalitesini yükseltmesi beklenir (24). Fakat hata oranının hesaplanması kolay değildir (24). Sadece bir eğitim kümesine dayalı hata oranı uygun bir tahmin sağlamaz (24). Öngörülen hata oranını tahmin etmenin bir yolu da, varsa, yeni ve ilave test örneklerinin ya da çapraz geçerlilik sınaması tekniklerinin kullanılmasıdır (24). Bu teknik başlangıçta mevcut olan örnekleri eşit boyutlu bloklara böler ve ağaç, her bir blok için, bu blok hariç olmak üzere, bütün örneklerden faydalanarak kurulup verilen bir örnekler bloğuyla test edilir (24). Karar ağacının budanmasındaki temel fikir, daha az karmaşık ve böylelikle daha kapsamlı bir ağaç oluşturmak için, mevcut eğitim ve test örnekleriyle, görülmeyen test örneklerinin doğru bir şekilde sınıflandırmasına katkıda bulunmayan alt ağaçların çıkarılmasıdır (24). Yinelemeli ayırma metodunun değiştirilebileceği iki yol vardır (24).

1. Bazı koşullar altında bir örnekler kümesinin daha fazla bölünmesine karar verilmesi (24). Durma kriteri genellikle χ^2 (ki-kare) testi gibi bazı istatistiksel testlere dayanmaktadır (24): Bölünmeden önce ve sonra sınıflandırma doğruluğunda belirgin farklar olmaması halinde, bir akım düğümü bir yaprak olarak gösterilir (24). Karar ayrım yapılmadan önce verilir, bu yüzden bu yaklaşım ön budama olarak adlandırılır (24).

2. Ağaç yapısının bir kısmının, seçilen doğruluk kriterlerinin kullanılmasıyla, geriye dönük olarak kaldırılması (24). Bu son budama prosesindeki karar, ağaç oluşturulduktan sonra verilir (24).

C4.5 son budama yaklaşımını izlemekte, fakat öngörülen hata oranını tahmin etmek için özel bir teknik kullanmaktadır (8). Bu metot kötümser budama olarak adlandırılmaktadır (8). Bir ağaçtaki her düğüm için, tahmin edilen üst güven limiti U_{cf}, binom dağılımı istatistik tablosu kullanılarak hesaplanır (8). U_{cf} parametresi verilen bir düğüm için T_i ve E'nin bir fonksiyonudur (8). C4.5, %25 varsayılan güven sınırını kullanır ve verilen her bir düğümdeki T_i için $U_{25\%}\left(\frac{|T_i|}{E}\right)$, düğüm yapraklarının güven aralığı ile karşılaştırır (8). Her bir yaprakta ağırlıklar olayların toplam sayısıdır (8). Bir alt ağaçtaki kök düğümün beklenen hatası, yapraklardaki için $U_{25\%}$ toplam ağırlıktan (alt ağaç için öngörülen hata) az olması halinde alt ağaç, budanan ağaçta yeni bir yaprak haline gelen kök düğümüyle değiştirilecektir (8).

C4.5'te budama yöntemini daha net açıklayabilmek için Quinlan (1993) tarafından yapılan bir çalışma örnek olarak verilecektir (8).

10 niteliğe sahip bir veri kümesi oluşturulsun (8). Değişkenlerin her biri 1 ve 0'lardan oluşan iki veri kümesi rasgele yaratılır (8). Sınıf, 'evet' ve 'hayır' değerlerinden oluşmak üzere ikili dallanma oluşturulur (8). 'Evet' 0.25 ağırlığına, hayır ise 0.75 ağırlığına sahiptir (8). Rasgele oluşturulan bin adet olay, 500 satırlık bir eğitim kümesi ve 500 satırlık bir test kümesi oluşturacak şekilde ayrılmıştır (8). Bu verilerden, C4.5'in başlangıçtaki ağaç oluşum rutiniyle, test olaylarında %35 hata oranına sahip, 119 düğümlük mantıksız bir ağaç meydana gelir (8). Bulunan hata oranı ağaçta oluşan tahmini hata oranının bile altındadır (8).

Karar ağacında başlangıç ağacından gelen dallanmanın kabul edilmesinin iki sakıncası vardır (8). Genellikle aşırı karmaşıktır ve basit bir ağaca göre çok daha büyük bir hata oranına sahip olabilir (8). Yukarıda belirtilen rasgele seçilen veri için, sadece 'hayır' yaprağı içeren bir ağaç, görülmeyen durumlarda %25 hata oranı verir (8). Ancak görüldüğü gibi her iki şıkkı içeren ağaç daha büyük bir hata oranı vermiştir (8).

Çoğunluk sınıfına ait olan durumların oranı, p≥ 0.5 (burada 'hayır')'dir (8). Eğer bir sınıflandırıcı tüm bu olayları bu çoğunluk sınıfına atarsa, sınıfın beklenen hata oranı (1-p) olur (8). Diğer taraftan sınıflandırıcı bir olayı, p olasılıklı bir çoğunluk sınıfına ve (1-p) olasılıklı diğer sınıflara atarsa, beklenen hata oranı aşağıdaki olasılıkların toplamı olur (8):

Çoğunluk sınıfına ait bir olayın diğer sınıfa atanması olasılığı, p x (1-p)

Diğer sınıfa ait bir olayın çoğunluk sınıfına atanması olasılığı, (1-p) x p

Bu değerlerin toplamı 2 x p x (1-p) olur. p en az 0.5 olduğu için, ki bu genellikle (1-p) den daha büyüktür, ikinci sınıflandırıcı daha büyük bir hata oranına sahip olacaktır (8). Söz konusu veri tabanı için tahmini hata oranı 2x0.75x0.25=%0.37 sonucuna ulaşır (8). Karar ağacı bu tahmini hata oranlarının çok altında bir performansa sahip olabilmelidir (8). Oysaki

rasgele seçilen bir veri yığınının oluşturduğu ağacın hata oranı bu tahmini hata oranına çok yakındır (8). Bu ise aşırı uyumun en büyük sorununun verinin söz konusu olayla ilgili olmaması veya bir şey ifade etmemesi olduğunu göstermiştir (8).

CART Algoritması

CART (Classification and Regression Trees) tekniği ID3 algoritmasında olduğu gibi en iyi dallara ayırma kriterini seçmek için entropiden yararlanır (16). En iyi ayırma kriterini belirlemek için ID3 ve C4.5'ten farklı bir formül kullanır. Algoritma sınıflandırma (Classification) ve regresyon ağacı üzerine dayalıdır (23). CART dallara ayırma kriterini hesaplarken kayıp verileri önemsemez (23). Bir CART ikili bir karar ağacıdır, bir düğümle bölünen ve düğümün 2 alt düğümle (ondan doğan) tekrar etmesi yapısı vardır (23). Başlangıç kök düğümü bütün örneklem bilgisini içerir (23). CART, dallanması sürecinde, tekrarlanan ikili bölümlemeye göre tahminleme yapar (48). CART'nin klasik doğrusal ve LR algoritmalarına göre potansiyel avantajı, parametrik istatistik varsayımlarına bağlı olmayan yani parametrik olmayan ve doğrusal olmayan bir metot olmasıdır (48). Bundan dolayı CART, belirli bir çıktıyı tahmin etmek için en önemli tahmin edici değişkenler arasındaki ilişkiyi göz önüne almadan çok sayıda değişken arasından, tahmin edici değişkeni seçebilir (48).

Regresyon ağacın büyüme süreci:

Ağacın büyümesindeki ana fikir; her bir düğümdeki tüm olabilir bölünme pozisyonları içinden bir bölünme seçmek ve bu seçtiğimiz bölünmeden doğacak bölünmenin "esnaf" olmasıdır (21). Bu algoritmada, sadece ikili ayrılmalar düşünülür (21). Böyle, her bir ayrılma bir sonuç değişkeninin değerine bağlıdır (21). Bütün olabilir ayrılmalar her bir tahmin edicinin ayrılmalarından meydana gelmektedir (21). Eğer X değişkeni isimsel kategorik (ordinal) değişken I kategoriden meydana gelmişse bu değişken için (2^{I-1}-1) tane ayrılma mümkündür (21). Eğer bu X değişkeni K tane farklı değere sahip sıralı kategorik veya sürekli değişken ise bu değişken üzerinde K-1 tane ayrılma yapılabilir (21). Bir ağaç X kök düğümünden büyümeye başlar ve aşağıdaki adımlar her bir düğüm için tekrarlanır (21).

1. Her bir tahmin edicinin en iyi bölünmesini bulmak: Her bir sürekli ve isimsel sıralı tahmin edici için en geniş aralıklıdan en dar aralığa sıralamak. Sıralanmış tahmin ediciler için, her bir verinin zirvesinden her bir bölünmeyi (eğer $x \leq v$ ise, v diye çağrılır, gözlem soldaki doğacak bölünmeye gider, aksi halde sağdakine gider)

denetleyip en iyi bölünme noktasına karar verir. En iyi bölünme noktası bir tanedir ve bu nokta bölünme kıstaslarını maksimize edecek şekilde böler. Her isimsel tahmin edici için, olası kategorik alt kümelerin hepsi (Eğer $x \in A$ ise, A diye çağrılır, gözlem ilk önce soldaki doğacak olan düğüme gider, aksi halde sağdakine gider) en iyi bölünmeyi bulmak için gözden geçirilir.

2. Düğümün en iyi bölünmesini bulmak: Birinci adımda bulunan en iyi bölünmeler arasından bölünme kıstaslarını maksimize edeni bulmak.
3. Eğer durma kuralları yetersiz ise, adım 2'de bulunan en iyi bölünmeyi kullanarak düğümü bölmek (21).

Bölünme kıstasları ve saflığın bozulması:

Bir bölünme kıstasını maksimize eden "t" düğümünde en iyi bölünme "s" seçilmiş olsun $\Delta_i(s,t)$. Bir düğüm için saflığı bozan ölçütler tanımlanmış ise saflıkta bir azalmaya karşılık gelen bölünme kriteridir (21): $\Delta I(s,t) = p(t)\Delta_i(s,t)$

Kategorik bağımlı değişken:

Eğer Y değişkeni kategorik ise; 3 tane bölünme kriteri mümkündür: Gini, Twoing ve Düzeltilmiş Twoing kriteri (21).

T düğümündeki olasılıklar : $p(j,t), p(t)$ ve $p(j/t)$ tahmini;

$$p(j,t) = \frac{\pi(j)N_{w,j}(t)}{N_{w,j}}$$

$$p(t) = \sum_j p(j,t)$$

$$p(j,t) = \frac{p(j,t)}{p(t)} = \frac{p(j,t)}{\sum_j p(j,t)}$$

Buradaki

$$N_{w,j} = \sum_{n \in \mathfrak{h}} W_n f_n I(y_n = 1), N_{w,j}(t) = \sum_{n \in \mathfrak{h}(t)} W_n f_n I(y_n = 1),$$

$I(a = b)$ gösterge fonksiyonu a=b ise 1 diğer durumlarda 0 değerini alan bir fonksiyondur.(21)

Gini kriteri:

Gini kriterine göre bir "t" düğümündeki "k" saflığın bozulması aşağıdaki gibidir;

$$i(t) = \sum_{i,j} C(i/j)p(i/j)p(j/t)$$

Gini bölünme kriterleri saflığın bozulmasını azaltması aşağıdaki gibi tanımlanır;

$$\Delta_i(s,t) = i(t) - p_L i(t_L) - p_R i(t_R)$$

Burada p_L ve p_R bir gözlemin sırasıyla soldan doğan düğüm t_L ve sağdan doğan düğüm t_R ye gönderilme olasılığıdır (21). Şu şekilde tahmin edilir $p_L = p(t_L)/p(t)$ ve $p(t_R)/p(t)$ (21).

Araştırmayı yapan kişinin belirlediği eklenmiş değerler, değiştirilmiş ön olasılıklar bunların (ön olasılıkların) yerine kullanılır (21). Değiştirilmiş ön olasılıkları kullanırken problem eklenmiş değerler yokmuş gibi görünürler (21). Değiştirilmiş ön olasılıklar (21):

$$\pi'(j) = \frac{C(j)\pi(j)}{\sum_j C(j)\,\pi(j)}, \text{burada } C(j) = \sum_j C(\,i/j).$$

Twoing kriteri:

$$\Delta i(s,t) = p_L p_R \left[\sum_j |p(j/t_L)) - p(j/t_R))|\right]^2$$

Düzeltilmiş twoing kriteri:

Y değişkeni sadece isimsel sıralı olduğunda düzeltilmiş Twoing kullanılır (21). Algoritması aşağıdaki gibidir (21):

1. Y'den ilk ayrılan sınıf $C = \{1, \dots, j\}$ daha sonra ikinci üstün sınıf C_1 ve$C_2 = C - C_2$ olarak bulunur, aynı şekilde $C_1 = \{1, \dots, j\}, j_i = 1, \dots . , j - 1$ şeklindedir (21).

2. İki sınıf ölçüsü kullanılır; $i(t) = p(C_1/t)p(C_2/t), \Delta i(s,t)$ 'yi maksimize eden bölünme $S^*(C_1)$ bulmak için

$$\Delta i(s,t) = i(t) - p_L i(t_L) - p_R i(t_R) - p_R i(t_R) = p_R p_L \left[\sum_{j \in C_1} \{p(j/t_L) - p(j/t_R)\}\right]^2$$

3. C_1'in üstün sınıfı C_1^* ,değeri $\Delta i(S^*(C_1), t)$'yi maksimize eder (21).

Sürekli bağımlı değişkenler:

Y sürekli olduğu zaman, bölünme kriteri $\Delta i(s,t) = i(t) - p_L i(t_L) - p_R i(t_R)$ En küçük kareler sapması kullanılır (21). Saflığın Bozulması;

$$i(t) = \frac{\sum_{n \in \hbar(t)} W_n f_n \left(y_n - \bar{y}(t)\right)^2}{\sum_{n \in \hbar(t)} W_n f_n}$$

Burada

$$p_L = N_w(t_L)/N_w(t),\ p_R = N_w(t_R)/N_w(t),$$

$$N_w(t) = \sum_{n \in \hbar(t)} W_n f_n\ , \bar{y}(t) = \frac{\sum_{n \in \hbar(t)} W_n f_n}{N_w(t)}$$

Kuralların durması:

Regresyon ağacının büyümesinin durup, durmaması ile ilgili kontrol kuralların durması kriteriyle tespit edilir. Kuralların durması kriterleri;

- Eğer bir durum saf ise; düğümdeki tüm gözlemler bağımlı değişkenlerin değerleriyle aynıdır.
- Bir düğümdeki tüm gözlemler her bir tahmin ediciyle aynı değerde ise; düğüm bölünmez.
- Son ağaç derinliği araştırmacının belirttiği maksimum ağaç derinlik limitine ulaşmış ise; ağacın büyüme süreci son bulur.
- Bir düğümün büyüklüğü araştırmacının belirttiği minimum düğüm büyüklüğünden daha az ise; düğüm bölünmez.
- Bir düğümün bölünmesi sonucu doğan düğümün büyüklüğü kullanıcının belirttiği doğan düğümlerin minimum büyüklüğünden küçük ise; bölünme olmaz.

- Bir t düğümünün en iyi bölünmesi olan s* ilerlemesi $\Delta I(s^*, t) = p(t)\Delta i(s^*, t)$ kullanıcının belirttiği minimum ilerlemeden daha küçük ise düğüm bölünmez (21).

CHAID Analizi

Karar ağaçları içinde CHAID kategorik değişkenlerdeki karmaşık etkileşim veya kombinasyonları bulan bir yöntemdir (21). Yöntem, ana kütleyi bağımlı değişkeni en önemli açıklayıcı değişkene göre alt guruplara veya bölümlere tekrarlı olarak ayırmaktadır (21). CHAID analizi, geniş veri kümelerini daha anlaşılır bir şekilde yorumlayabilmek için, sınıflandırma ölçme düzeyinde ölçülmüş, bir bağımlı değişkeni, en iyi açıklayabilecek detaylı alt kümelere böler (42). Bu bölünme işlemini yaparken, tahmin edicilere ait kategorileri yeniden kategorileştirerek, her alt kümeye ayırma işlemini bağımsız olarak gerçekleştirir (42). CHAID analizi, oluşturulan eşleşmedeki değişkenler arasındaki ilişkileri belirlediği gibi, sonuçlarını bir ağacın dalları şeklinde anlaşılır bir şeklinde ifade eden yöntemdir (21). CHAID (Otomatik Ki-Kare Etkileşim Belirleme) tekniği kategorik bağımlı değişkenler için tasarlanmış AID analizinin bir uzantısıdır (1). Bu analizde amaç veriyi daha homojen alt gruplara bölmektir (1). Araştırmacı oldukça homojen bir veri kümesi ile çalışmak ister (1). Büyük çapta bir veri kümesinin homojen bir alt gruba indirgenmesi problemi demek; bağımlı değişkeni mümkün olduğunca tutumlu bir şekilde açıklayan diğer değişkenleri ve bunlarla ilgili verileri ortaya koymak demektir (20). İşte CHAID analizi, kategorik değişkenlere ilişkin veri kümesini, bağımlı değişkeni en iyi açıklayacak şekilde detaylı homojen alt gruplara böler (20). Bu alt gruplar küçük tahmin edici gruplardan oluşur (20). Seçilen tahmin ediciler daha sonraki ileri analizlerde bağımlı değişkenin tahmininde kullanılacaktır (20). CHAID, regresyon problemlerinde kullanılabileceği gibi karar ağaçlarının oluşturulmasında etkilidir (20). Değişkenler arasındaki ilişki lineer yapıdan daha karmaşık ise veride gizli olan bu ilişkiyi bulmak için verinin belli kısımlarını eleme yöntemi olan CHAID kullanılır (20). "Ki-Kare" ismini almasının nedeni algoritmasında birçok çapraz tablonun kullanılması ve istatistiksel önem oranlarıyla çalışmasıdır (20).

CHAID analizi;

- Sınıflama ölçme düzeyinde ölçülmüş bir bağımlı değişkeni en iyi şekilde açıklamak için kullanılır,
- Açıklayıcı değişkenler sınıflayıcı, sıralayıcı ve aralıklı ölçek ile ölçülmüş olabilir,
- Kayıp verileri yeni bir kategori gibi davranır ve bu kategoriyi p-değeri hesaplamalarına dâhil eder,

- Kategorileri sıralanabilen ya da sıralanamayan, açıklayıcı değişkenlerin yer aldığı veri kümesini, bağımlı değişkene göre detaylı alt kümelere böler,
- Bu bölünme işlemini gerçekleştirirken, açıklayıcı değişkenlere ait kategorileri, bağımsız olarak yeniden düzenler, yani kategorileştirir.
- Daha sonraki her bölünmeyi yeniden bağımsız olarak gerçekleştirir (20).

Yani CHAID analizi, çok kategorili değişkenlerin yer aldığı büyük bir veri kümesini, benzer kategorileri birleştirerek, önemli sayılan değişkenlere göre bölerek, bir bakıma önceki durumuna oranla özet şekilde tanımlamış olur (20). Her bir açıklayıcı değişken için kategorilerin anlamlı bir şekilde birleştirilmesinden sonra, bağımlı değişkene göre kontenjans tabloları oluşturularak, Bonferroni p değerleri ile χ^2 istatistikleri hesaplanır (20). Açıklayıcı değişkenler birbiri ile karşılaştırılıp, en küçük Bonferroni p değerine sahip olan açıklayıcı değişkenin kategorilerine göre, veriler alt gruplara ayrılır (20).

CHAID algoritması, sadece nominal veya sıralı kategori belirleyicilerini kabul eder (46). Belirleyiciler sürekli olduğunda, bir sonraki algoritmayı kullanmadan önce sıralı belirleyicilere dönüştürülür (46). Her bir tahmin edici değişken X için, anlamlı olmayan kategorileri birleştirir (46). Her bir son X kategorisi, eğer X, düğümü bölmek için kullanılırsa, küçük bir düğüm ile sonlanır (46). Birleştirme adımı ayrıca, bölücü adımlarda kullanılan düzenlenmiş p değerini hesaplar (46).

CHAID analizinde her bir açıklayıcı değişken için en iyi bölünme bulunur (20). Daha sonra açıklayıcı değişkenler en iyi seçilene kadar karşılaştırılır ve seçilen en iyi açıklayıcı değişkene göre yeniden bölünmeler yapılır (20). Tüm alt bölünmeler bağımsız olarak yeniden analiz edilir (20). Her bir açıklayıcı değişken kategorilerini izin verildiği mümkün bölünmeler gerçekleştirerek χ^2 testinden önem derecesine göre kontenjans tabloları oluşturulur (20). Buradan yola çıkarak CHAID analizi χ^2 istatistiklerini, Bondferroni yaklaşımını ve kategori birleştirme algoritmasını kullanarak araştırmacının ağaç diyagramı ile en önemli açıklayıcı değişkenleri ve bağımlı değişken ile olan etkileşimleri elde etmesini sağlar (20).

CHAID analizinin algoritması:

Bağımlı değişken kategori sayısı $d \geq 2$ olsun (20). Analiz edilecek olan belirli bir açıklayıcı değişken $c \geq 2$ sayıda kategoriye sahip olsun (20). Analizdeki amaç, c x d kontenjans tablosunu açıklayıcı değişkenindeki uygun kategorileri birleştirme yolu ile en anlamlı j x d tablosuna indirgemektedir (20). Kavramsal olarak ilk olarak $Tj^{(i)}$ istatistiğini hesaplarız (20). $Tj^{(i)}$ j x d tablosunu oluşturmada i. metot için, χ^2 istatistiğidir (20).

(J=2,3,4,…,c; i'nin değişim aralığı açıklayıcı değişkenin tipine bağlıdır.) $Tj^{(*)}$= $maxiTj^{(i)}$ ise en iyi j x d tablo için, χ^2 istatistiği elde edilmiş olur (20). Yani, en önemli $Tj^{(i)}$ seçilir (20).

Monotonik ya da dichotomous serbest açıklayıcı değişkenin varlığında $Tj^{(i)}$ Fisher metoduna göre bulunabilir (20). Bu dinamik program c^2 hesaplarına dayanır (20). $d \geq 3$ve açıklayıcı değişken sıralı kategorilere sahip değilse Fisher metodundan yararlanılamaz (20). Dreyfus 1977'de dinamik programlarda standart uygulamaların permütasyon tipi problemlerde uygulanabilir olduğunu göstermiştir (20). Bu çözüm ise 2^c kadardır (20).

Algoritma 3 aşamadan oluşmaktadır; birleştirme, dağıtma ve durdurma (1).

a- Birleştirme:

1. Adım: Her bir açıklayıcı değişken için sırasıyla, açıklayıcı değişkenin kategorileri ile bağımlı değişkenin kategorilerinin çapraz tablosu bulunur ve adım 2 ve 3 uygulanır (1).

2. Adım: Sadece açıklayıcı değişkenin tipi tarafından belirlenen uygun çiftler göz önüne alınarak, 2 x d alt tablosunda anlamlılığı düşük alan açıklayıcı değişken kategori çiftleri bulunur (1). Eğer önem derecesi kritik bir değere ulaşmıyorsa, bu iki kategori birleştirilir (1). Ve bu birleşim tek bir kategori olarak ele alınır ve bu adım tekrarlanır (1). Bu işlem açıklayıcı değişkenin kendi içindeki birleşmeleri anlamsız oluncaya kadar devam eder (1).

3. Adım: Açıklayıcı değişkenin tipi tarafından oluşturulan ve orijinal kategorilerin 3 veya daha fazlasının birleştirilmesi ile meydana gelen; her bir bileşik kategori için, birleşmenin tekrar ayrılabileceği en önemli ikili bölünme bulunur(1). Eğer önem derecesi kritik değerin üzerindeyse, bölünme gerçekleştirilir ve 2. adıma dönülür (1).

b- Dağıtma:

4. Adım: Optimal bir şekilde birleştirilmiş olan, her bir açıklayıcı değişken için önem derecesi hesaplanarak, en büyük önem derecesine sahip olan, diğerlerinden ayrılır (1). Eğer bu önem derecesi, verilen kriter değerlerde büyük ise, veri kümesi için seçilen açıklayıcı değişkenin birleştirilmiş kategorilerine göre alt guruplarına bölünür (1).

c- Durdurma:

5. Adım: Verinin analiz edilememiş her bir gurubu için, birinci adıma dönülür bu adımda en az sayıda gözleme sahip olan guruplar göz ardı edilebilir (1).

Açıklayıcı değişkenler önemliliği:

Algoritmanın 4. adımında, indirgenmiş olan kontenjans tablosunun önem derecesinin, test edilmesi gerekir (1). Eğer orijinal kontenjans tablosunda herhangi bir indirgenme yoksa χ^2 testi kullanılabilir (1). χ^2 testi açıklayıcı değişkenin kategori sayısına bağlıdır, aksi halde çok dikkatli bir şekilde uygulanmalıdır (1). Kesin sonuçlar bilinmiyorsa ya da orijinal olasılık tablosu indirgenmemiş ise Benforroni sonuçlarını kullanılması tercih edilir (1).

Kategorik değişkenlere ilişkin veri kümesini ve bağımlı değişkeni en iyi açıklayabilecek değişkenleri ayrıntılı homojen alt gruplara bölen Chaid çözümlemesi en iyi tahmin sonucunu elde etmek için başlangıç değişkenlerini yeniden kategorileştirir (41). Benzer kategorileri birleştirilir ve değişkeler arasında daha fazla birleştirme işlemi gerçekleşemeyinceye kadar devam eder (41). Değişkenlerin birleşmeye uygun olup olmadığına Bonferroni düzeltilmiş p değeri kullanılarak karar verilir (41).

Orijinal olasılık tablosunun indirgenmesi; her bir açıklayıcı değişken için, kendi içinde kategorileri anlamlı bir şekilde birleştirilip, en iyi bölünmenin bulunmasından sonra, bağımlı değişkene göre kontenjans tablosunun oluşturulması demektir (1). Daha sonra χ^2 ile Benforroni düzeltilmiş p değerleri hesaplanır (1).

LOJİSTİK REGRESYON (LR)

Çok değişkenli istatistik yöntemlerinden biri olan LR, sınıflama ve atama işleminde kullanılabilen bir regresyon yöntemidir (32). Bağımlı değişkenin kesikli, bağımsız değişkenlerin hem kesikli; hem de sürekli olduğu durumlarda uygulanabilen, normal dağılım ve süreklilik ön koşulları bulunmayan bir tekniktir (32). LR ile bağımlı değişken üzerinde bağımsız değişkenlerin etkili olasılık olarak belirlenen risk faktörlerinin olasılık olarak belirlenmesi sağlanır (32). LR; kategorik ve ikili (binary, dichotomous), üçlü ve çoklu kategorilerde gözlendiği durumlarda bağımlı değişkenin bağımsız değişkenler ile olan neden sonuç ilişkisini belirlemede yararlanılan bir yöntemdir (32). LR ve sınıflandırma ağaçları, bağımlı değişkenin herhangi bir varsayımı olmadan kategorik bağımsız değişkenlerin sınıf ilişkilerini tespit ederken kullanılmaktadır (47). Bu metotlar, genellikle öğrenme uygulamalarında, bilgisayar bilimlerinde veri madenciliğinde ve sınıflandırma modellerinde kullanılmaktadır (47).

Bağımlı değişkenin iki ya da çok sınıflı kesikli değişken olması durumunda kullanılabilecek modeller çok çeşitlidir (32). Bu modellerden doğrusal olasılık modeli, lojit ve probit modeller arasında en fazla tercih edilen yöntem LR'dır (32). LR, normallik

varsayımının bozulması nedeniyle doğrusal regresyon analizine alternatif olmaktadır (32). LR'da bağımsız değişkenler ile iki ya da daha çok sınıflı kategorik bağımlı değişken arasındaki ilişkinin tanımlanması için matematiksel modelleme yapmak amaçlanmaktadır (32).

$$E(y_i) = P(y_i = 1) = \sum_{k=0}^{p} \beta_k x_{ik} \qquad i = 1, \dots, n$$

Modellinde olasılık değerleri üzerinde $P/1-P$ dönüşümü yapılarak bağımlı değişkenin sınırları 0,+∞ yapılmakta, daha sonra ise bu oran değerinin logaritması alınarak bağımlı değişkenin sınırları -∞, +∞ yapılmaktadır (32). Bu dönüşümlerden sonra elde edilen yeni fonksiyon:

$$E(y_i) = P(y_i = 1) = L_i = \ln\left(\frac{P_i}{1-P_i}\right) = \sum_{k=0}^{p} \beta_k x_{ik}$$

olarak yazılabilir. Bu modele "lojistik model" ya da kısaca "lojit" denmektedir (32). Ayrıca kullanılan $\ln(\frac{P}{1-P})$ dönüşümü de "lojit dönüşüm" adını almaktadır (32). Lojistik fonksiyonun elde edildiği modelde kullanılan P_i olasılık değeri ise:

$$P_i = \frac{\exp\left(\sum_{k=0}^{p} \beta_k x_{ik}\right)}{1 + \exp\left(\sum_{k=0}^{p} \beta_k x_{ik}\right)}$$

biçiminde tanımlanmaktadır (32). Bu modelde bağımlı değişkenin iki sınıflı olması sebebiyle hata terimi ε;

$$y_i = \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij} + \varepsilon_i = 0 \;\Rightarrow\; \varepsilon_i = -\beta_0 - \sum_{j=1}^{k} \beta_j x_{ij}$$

$$y_i = \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij} + \varepsilon_i = 1 \;\Rightarrow\; \varepsilon_i = 1 - \beta_0 - \sum_{j=1}^{k} \beta_j x_{ij}$$

değerlerini almaktadır (32). Hata terimlerine ilişkin daha önce verilenlerden yola çıkarak;

$E(\varepsilon_i)=Pr(y_i=0)(-\beta_0 + \sum_{j=1}^{k} \beta_j x_{ij})+Pr(y_i=1)(1-\beta_0 + \sum_{j=1}^{k} \beta_j x_{ij})$

$E(\varepsilon_i)=0$ ve

$V(\varepsilon_i)=E(\varepsilon_i^2)$

$= \Pr(y_i = 0)(-\beta_0 + \sum_{j=1}^{k} \beta_j x_{ij})^2 + \Pr(y_i=1)(1 - \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij})^2$

$=(1 - \Pr(y_i = 1))(\Pr(y_i = 1))^2 + \Pr(y_i = 1)(1 - \Pr(y_i = 1))^2$

$=\Pr(y_i=1)(1-\Pr(y_i=1))$

$=P_i(1-P_i)$

varsayımları sağlanmaktadır (32). Yani hata terimi 0 ortalama ve $P(1-P)$ varyanslıdır (32). Hata terimi bu parametrelerle binom dağılımı olup, analiz de bu teorik temele dayanmaktadır (32). Lojistik modele ilişkin varsayımlar kısaca şöyledir:

1) $y_i \in (0,1)$

2) $P(y_i = 1|x_i) = P_i$

3) $y_i,\ldots,y_n$ değerleri istatistiksel olarak bağımsızdır,

4) Bağımsız değişkenler olan x_k'lar birbirinden bağımsızdır.

Ayrıca modelin bağımlı değişkeninin sınırlarını genişletmek için kullanılan $\text{In}(\frac{P}{1-p})$ lojit dönüşümünün de bazı önemli özellikleri şunlardır:

- P arttıkça lojit (P) de artar,
- P, 0-1 arasında iken lojit (P) tüm reel sayı değerlerini alır,
- Eğer $P < 0,5$ ise lojit $(P) < 0$,
- Eğer $P > 0,5$ ise lojit $(P) > 0$ ve
- Eğer $P = 0,5$ ise lojit $(P) = 0$'dır (32).

Bağımsız değişkenler üzerine herhangi bir kısıtlama getirilmeden LR analizinde bağımsız değişkenlerin durumuna göre farklı modeller kullanılabilir (32). Bu modeller:

a) Bağımsız değişkenlerin tümü kesikli ise;

$$\text{In}\frac{P_i}{1-P_i} = \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij}$$

b) Bağımsız değişkenlerin tümü sürekli ise $\Pr(x_1,\ldots,x_p)$ bağımsız değişken üzerinde koşullu başarı olasılığı olmak üzere lojistik model;

$$\text{In}\frac{\Pr(x_1,\ldots,x_p)}{1-\Pr(x_1,\ldots,x_p)} = \beta_0 + \sum_{j=1}^{k} \beta_j x_{ij}$$

c) Bağımsız değişkenlerin bazılarının sürekli bazılarının kesikli olması durumunda çok değişkenli frekans dağılımı başarı durumu için $f_1(x_1,\ldots,x_p)$ ve başarısızlık durumu için $f_0(x_1,\ldots,x_p)$ biçiminde tanımlanmış iken lojistik model;

$$\ln \frac{\Pr(x_1, \ldots, x_p)\ f_{1(x_1,\ldots,x_p)}}{(1 - \Pr(x_1, \ldots, x_p))\ f_0(x_1, \ldots, x_p)} = \beta_0 + \sum_{j=1}^{k} \beta_j\, x_{ij}$$

olarak tanımlanmaktadır (32).

Burada β katsayıları, gözlemleri f_0 ve f_1 fonksiyonlarına karşılık gelecek biçimde ayırma özelliğine sahip parametre değerleridir (32). Parametre tahmin değerleri ise en çok olabilirlik, yeniden ağırlıklandırılmış en küçük kareler ve minimum lojit Ki-Kare yöntemleri ile hesaplanır (32). Söz konusu LR modellerine ait fonksiyonlar süreklidir ve x bağımsız değişkeni ile β parametre değerleri ne olursa olsun olasılık 0 ile 1 arasında değerler almaktadır (32).

GEREÇ VE YÖNTEMLER

ARAŞTIRMANIN TÜRÜ

Araştırma bir simülasyon çalışmasıdır.

ARAŞTIRMANIN YAPILDIĞI YER VE ZAMAN

Araştırma Mayıs 2012 – Aralık 2013 tarihleri arasında Trakya Üniversitesi Tıp Fakültesi Biyoistatistik ve Tıbbi Bilişim Anabilim Dalında yürütülmüştür.

ARAŞTIRMA VERİLERİ

Bağımlı değişken olarak iki kategoriden oluşan değişken, bağımsız değişkenler olarak 10 farklı bağımsız değişken 3 farklı yapıdadır. Bunlar 10 bağımsız değişkenin tümü kategorik, 5 kategorik, 5 sürekli ve 10 bağımsız değişkenden hepsinin sürekli olduğu şekilde olmak üzere 30, 100 ve 1000'er denemelik veri türetilerek ulaşılmıştır. 1000'er denemelik veri türetilmesinde kullanılan parametreler (Tablo 1-7)'de gösterilmiştir. 30 ve 100 denemelik veri türetiminde de aynı parametreler kullanılmıştır.

Tablo 1. Bağımsız değişkenlerin tümünün kategorik olduğu durumda veri türetilmesinde kullanılan parametreler.

Hasta Grubu	Kontrol Grubu
x1=rbinom(1000, 1, 0.65)	x1=rbinom(1000, 1, 0.25)
x2=rbinom(1000, 1, 0.55)	x2=rbinom(1000, 1, 0.30)
x3=rbinom(1000, 1, 0.75)	x3=rbinom(1000, 1, 0.35)
x4=rbinom(1000, 1, 0.45)	x4=rbinom(1000, 1, 0.30)
x5=rbinom(1000, 1, 0.65)	x5=rbinom(1000, 1, 0.35)
x6=rbinom(1000, 1, 0.50)	x6=rbinom(1000, 1, 0.15)
x7=rbinom(1000, 1, 0.45)	x7=rbinom(1000, 1, 0.30)
x8=rbinom(1000, 1, 0.70)	x8=rbinom(1000, 1, 0.35)
x9=rbinom(1000, 1, 0.50)	x9=rbinom(1000, 1, 0.20)
x10=rbinom(1000, 1, 0.55)	x10=rbinom(1000, 1, 0.30)

Tablo 2. Bağımsız değişkenlerin 5 kategorik 5 sürekli (normal dağılım) olduğu durumda veri türetilmesinde kullanılan parametreler.

Hasta Grubu	Kontrol Grubu
x1=rbinom(1000, 1, 0.65)	x1=rbinom(1000, 1, 0.60)
x2=rbinom(1000, 1, 0.55)	x2=rbinom(1000, 1, 0.50)
x3=rbinom(1000, 1, 0.75)	x3=rbinom(1000, 1, 0.70)
x4=rbinom(1000, 1, 0.45)	x4=rbinom(1000, 1, 0.45)
x5=rbinom(1000, 1, 0.65)	x5=rbinom(1000, 1, 0.60)
x6=rnorm(1000, 80, 9)	x6=rnorm(1000, 75, 7)
x7=rnorm(1000, 90, 12)	x7=rnorm(1000, 85, 9)
x8=rnorm(1000, 120, 10)	x8=rnorm(1000, 110, 8)
x9=rnorm(1000, 65, 6.5)	x9=rnorm(1000, 60, 5.5)
x10=rnorm(1000, 85, 8)	x10=rnorm(1000, 80, 6)

Tablo 3. Bağımsız değişkenlerin 5 kategorik 5 sürekli (f dağılım) olduğu durumda veri türetilmesinde kullanılan parametreler.

Hasta Grubu		Kontrol Grubu
x1=rbinom(1000, 1, 0.65)		x1=rbinom(1000, 1, 0.60)
x2=rbinom(1000, 1, 0.55)		x2=rbinom(1000, 1, 0.50)
x3=rbinom(1000, 1, 0.75)		x3=rbinom(1000, 1, 0.70)
x4=rbinom(1000, 1, 0.45)		x4=rbinom(1000, 1, 0.45)
x5=rbinom(1000, 1, 0.65)		x5=rbinom(1000, 1, 0.60)
x6=rf(1000, 1, 10,30)		x6=rf(1000, 1, 10,13)
x7=rf(1000, 2, 15,15)		x7=rf(1000, 2, 4,15)
x8=rf(1000, 3, 20,20)		x8=rf(1000, 3, 6,11)
x9=rf(1000, 3, 3,40)		x9=rf(1000, 2, 3,10)
x10=rf(1000, 3, 5,25)		x10=rf(1000, 3, 5,12)

Tablo 4. Bağımsız değişkenlerin 5 kategorik 5 sürekli (3 sürekli f dağılım, 2 sürekli normal dağılım) olduğu durumda veri türetilmesinde kullanılan parametreler.

Hasta Grubu		Kontrol Grubu
x1=rbinom(1000, 1, 0.65)		x1=rbinom(1000, 1, 0.60)
x2=rbinom(1000, 1, 0.55)		x2=rbinom(1000, 1, 0.50)
x3=rbinom(1000, 1, 0.75)		x3=rbinom(1000, 1, 0.70)
x4=rbinom(1000, 1, 0.45)		x4=rbinom(1000, 1, 0.45)
x5=rbinom(1000, 1, 0.65)		x5=rbinom(1000, 1, 0.60)
x6=rf(1000, 1, 10,30))		x6=rf(1000, 1, 10,13)
x7=rf(1000, 2, 15,15))		x7=rf(1000, 2, 4,15)
x8=rf(1000, 3, 20,20))		x8=rf(1000, 3, 6,11)
x9=rnorm(1000, 65, 6.5)		x9=rnorm(1000, 60, 5.5)
x10=rnorm(1000, 85, 8)		x10=rnorm(1000, 80, 6)

Tablo 5. Bağımsız değişkenlerin tümü sürekli (normal dağılım) olduğu durumda veri türetilmesinde kullanılan parametreler.

Hasta Grubu	Kontrol Grubu
x1=rnorm(1000, 50, 15)	x1=rnorm(1000, 45, 15)
x2=rnorm(1000, 150, 45)	x2=rnorm(1000, 140, 40)
x3=rnorm(1000, 125, 32)	x3=rnorm(1000, 120, 29)
x4=rnorm(1000, 12, 3)	x4=rnorm(1000, 9, 2)
x5=rnorm(1000, 200, 50)	x5=rnorm(1000, 170, 35)
x6=rnorm(1000, 80, 20)	x6=rnorm(1000, 75, 15)
x7=rnorm(1000, 90, 12)	x7=rnorm(1000, 85, 10)
x8=rnorm(1000, 120, 10)	x8=rnorm(1000, 111, 10)
x9=rnorm(1000, 65, 15)	x9=rnorm(1000, 60, 12.5)
x10=rnorm(1000, 85, 21)	x10=rnorm(1000, 80, 15)

Tablo 6. Bağımsız değişkenlerin tümü sürekli (f dağılım) olduğu durumda veri türetilmesinde kullanılan parametreler.

Hasta Grubu	Kontrol Grubu
x1=rf(1000, 8, 13,10)	x1=rf(1000, 5, 9,8)
x2=rf(1000, 5, 15,12)	x2=rf(1000, 4, 11,10)
x3=rf(1000, 2, 10,20)	x3=rf(1000, 2, 8,18)
x4=rf(1000, 4, 14,25)	x4=rf(1000, 3, 12,22)
x5=rf(1000, 7, 18,30)	x5=rf(1000, 5, 16,28)
x6=rf(1000, 1, 10,28)	x6=rf(1000, 1, 9,25)
x7=rf(1000, 2, 15,15)	x7=rf(1000, 2, 10,15)
x8=rf(1000, 3, 17,20)	x8=rf(1000, 3, 15,11)
x9=rf(1000, 3, 3,40)	x9=rf(1000, 2, 3,10)
x10=rf(1000, 3, 5,25)	x10=rf(1000, 3, 5,15)

Tablo 7. Bağımsız değişkenlerin tümü sürekli (5 sürekli değişken f, 5 sürekli değişken normal dağılım) olduğu durumda veri türetilmesinde kullanılan parametreler.

Hasta Grubu		**Kontrol Grubu**
x1=rnorm(1000, 50, 15)		x1=rnorm(1000, 45, 15)
x2=rnorm(1000, 150, 45)		x2=rnorm(1000, 145, 40)
x3=rnorm(1000, 125, 32)		x3=rnorm(1000, 120, 29)
x4=rnorm(1000, 12, 3)		x4=rnorm(1000, 9, 4)
x5=rnorm(1000, 200, 50)		x5=rnorm(1000, 185, 35)
x6=rf(1000, 1, 10,28)		x6=rf(1000, 1, 10,20)
x7=rf(1000, 2, 15,15)		x7=rf(1000, 2, 10,15)
x8=rf(1000, 3, 20,20)		x8=rf(1000, 3, 10,11)
x9=rf(1000, 3, 3,40)		x9=rf(1000, 2, 3,10)
x10=rf(1000, 3, 5,25)		x10=rf(1000, 3, 5,15)

ARAŞTIRMADA KULLANILAN İSTATİSTİKSEL YÖNTEMLER

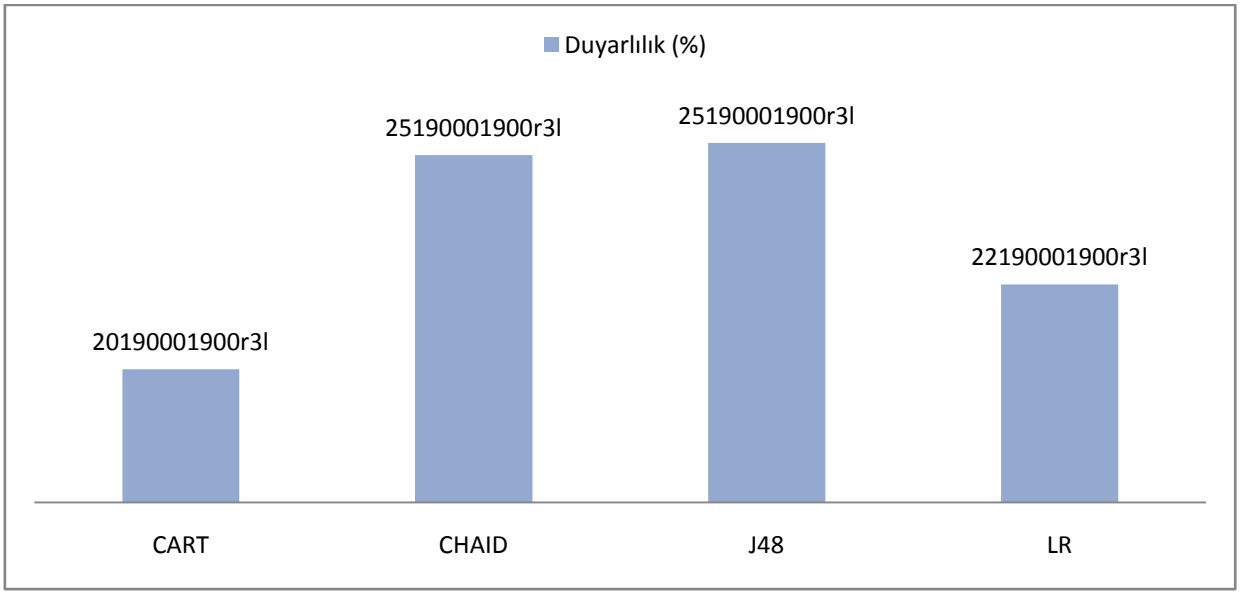
Türetilen verilerin analizinde CART, CHAID, J48 ve LR analizi yöntemleri kullanılmıştır. Bu yöntemlerin performanslarının karşılaştırılmasında duyarlılık (sensitivity), özgüllük (specificity), pozitif kestirim değeri, negatif kestirim değeri ve doğruluk oranlarının yanı sıra ROC analizi yönteminden de yararlanılmıştır.

BULGULAR

Tümü kategorik yapıda olan bağımsız değişkenler için 30 denemelik simülasyon çalışması sonuçlarına ilişkin duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC oranları (Tablo 8)'de gösterildi. Bu sonuçlara göre, dört yöntem arasında en düşük doğruluk oranı CART (%80) algoritmasında gözlenirken en yüksek doğruluk oranı J48 (%86,1) algoritmasında gözlenmiş olup bu orana en yakın doğru sınıfa atayabilme değerini ise CHAID algoritmasının aldığı bulunmuştur.

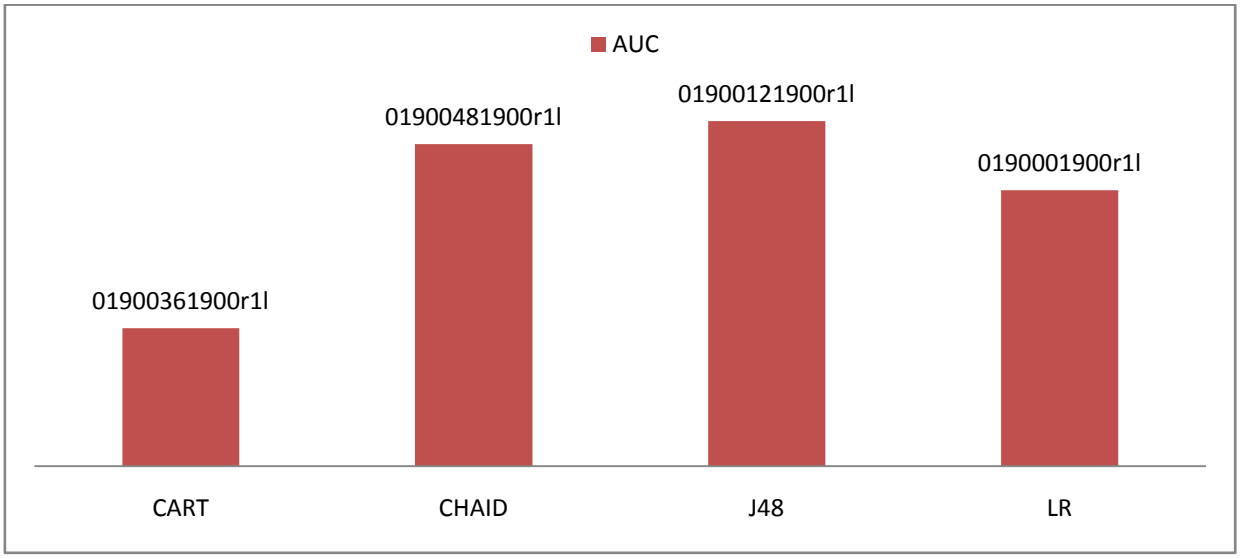
Tablo 8. Bağımsız değişkenlerin tümü kategorik olduğu durumda yöntemlere göre sınıflandırma sonuçları (30 deneme).

		CART	CHAID	J48	LR
30 deneme	Duyarlılık (%)	80,3	85,6	85,9	82,4
	Özgüllük (%)	79,7	83,4	86,4	83,4
	PKD (%)	79,9	83,8	86,3	83,4
	NKD (%)	80,3	85,3	86,0	83,1
	Doğruluk (%)	80,0	84,5	86,1	83,6
	AUC	0,84	0,92	0,93	0,90
	AUC'nin Standart Hatası	0,00901	0,00577	0,00678	0,00565



Sekil 3. Bağımsız değişkenlerin tümü kategorik olduğu durumda yöntemlere göre duyarlılık değerleri (30 deneme)

Bağımsız değişkenlerin tümünün kategorik olduğu 30 denemelik çalışma sonuçlarında en yüksek duyarlılık oranın (%85,9) ile J48 algoritmasında gözlenirken en düşük duyarlılık oranı CART yönteminde (%80,3) gözlenmiştir (Şekil 3).



Sekil 4. Bağımsız değişkenlerin tümü kategorik olduğu durumda yöntemlere göre eğri altında kalan alan (AUC) değerleri (30 deneme)

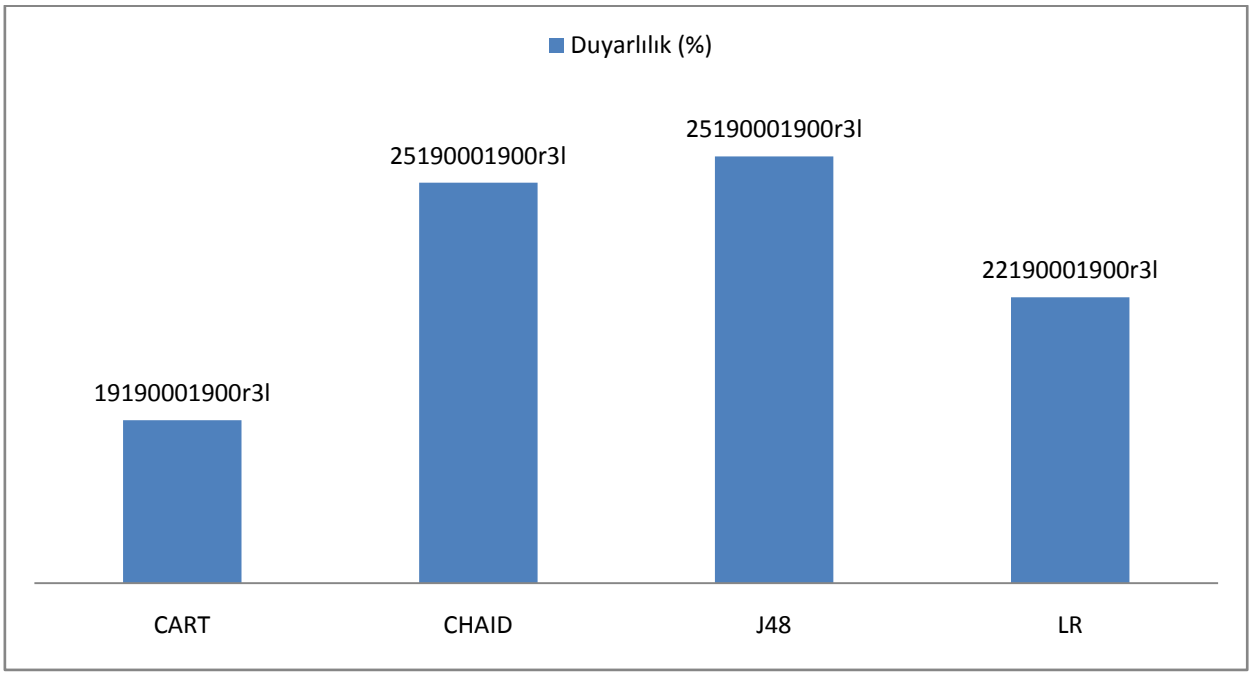
Bağımsız değişkenlerin tümünün kategorik olduğu durumda 30 denemelik çalışma sonuçlarına göre en yüksek AUC (eğri altında kalan alan) değeri J48 yönteminde (0,93) gözlenirken en düşük AUC değeri CART yönteminde (0,84) gözlenmiştir (Şekil 4).

Tümü kategorik yapıda olan bağımsız değişkenler için 100 denemelik simülasyon çalışması sonuçlarına ilişkin duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC oranları

(Tablo 9)'da gösterildi. Bu sonuçlara göre, dört algoritma arasında en düşük duyarlılık oranı (%79,7) CART algoritmasında gözlenirken diğer iki algoritmanın duyarlılık oranlarının birbirine yakın değerler (J48: %85,7; CHAID: %85,1) aldığı bulunmuştur.

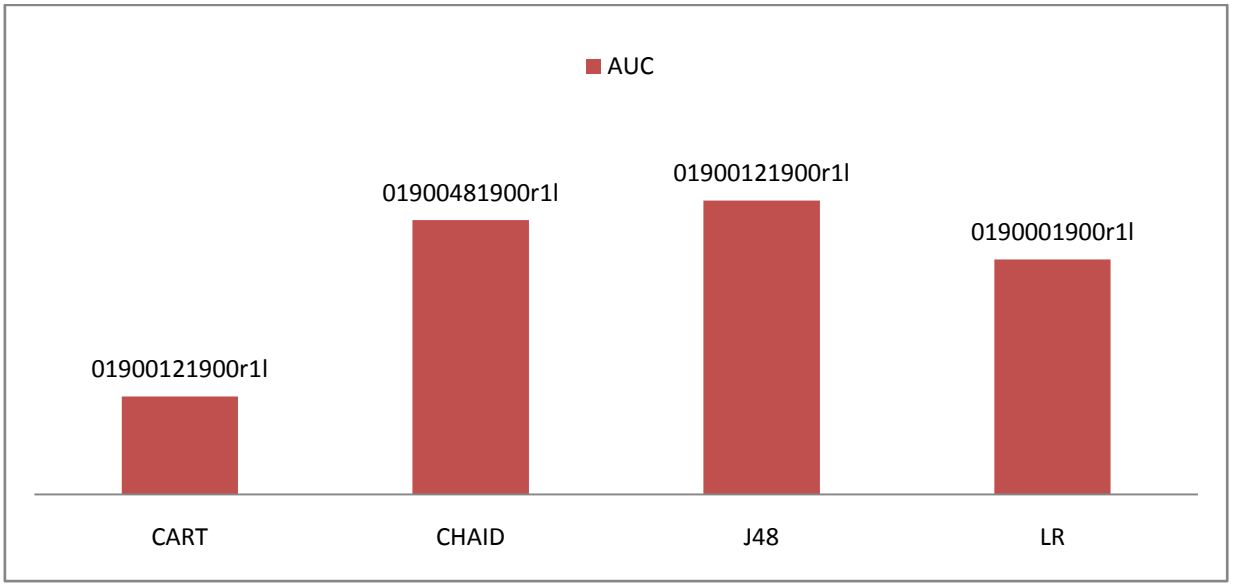
Tablo 9. Bağımsız değişkenlerin tümü kategorik olduğu durumda yöntemlere göre sınıflandırma sonuçları (100 deneme).

		CART	CHAID	J48	LR
100 deneme	Duyarlılık (%)	79,7	85,1	85,7	82,5
	Özgüllük (%)	80,0	83,8	86,4	83,5
	PKD (%)	80,0	84,1	86,5	83,4
	NKD (%)	79,9	85,0	85,8	83,1
	Doğruluk (%)	79,8	84,4	86,2	83,2
	AUC	0,83	0,92	0,93	0,90
	AUC'nin Standart Hatası	0,009	0,006	0,007	0,006



Sekil 5. Bağımsız değişkenlerin tümü kategorik olduğu durumda yöntemlere göre duyarlılık değerleri (100 deneme)

Bağımsız değişkenlerin tümünün kategorik olduğu durumda 100 denemelik çalışma sonuçlarında görüldüğü gibi en yüksek duyarlılık oranı (%85,7) ile J48 yönteminde, en düşük duyarlılık oranı CART yönteminde (%79,7) gözlenmiştir (Şekil 5).



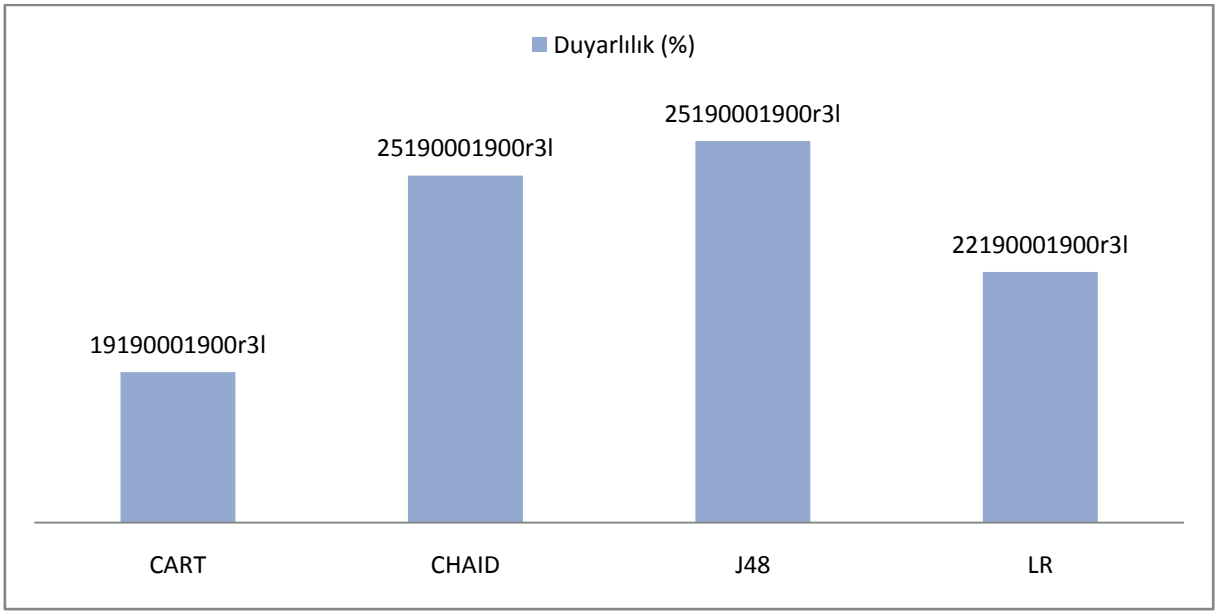
Sekil 6. Bağımsız değişkenlerin tümü kategorik olduğu durumda yöntemlere göre eğri altında kalan alan (AUC) değerleri (100 deneme)

Bağımsız değişkenlerin tümünün kategorik olduğu 100 denemelik çalışma sonucunda grafikte de görüldüğü gibi en yüksek AUC değeri J48 yönteminde (0,93), en düşük AUC değeri CART yönteminde (0,83) gözlenmiştir (Şekil 6).

Tümü kategorik yapıda olan bağımsız değişkenler için 1000 denemelik simülasyon çalışması sonuçlarına ilişkin duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC oranları (Tablo 10)'da gösterildi. Bu sonuçlara göre, dört algoritma arasında en düşük duyarlılık oranı CART yönteminde (%79,9) gözlenirken diğer iki yöntemin duyarlılık oranlarının birbirine yakın değerler (J48: %85,9; CHAID: %85) aldığı bulunmuştur.

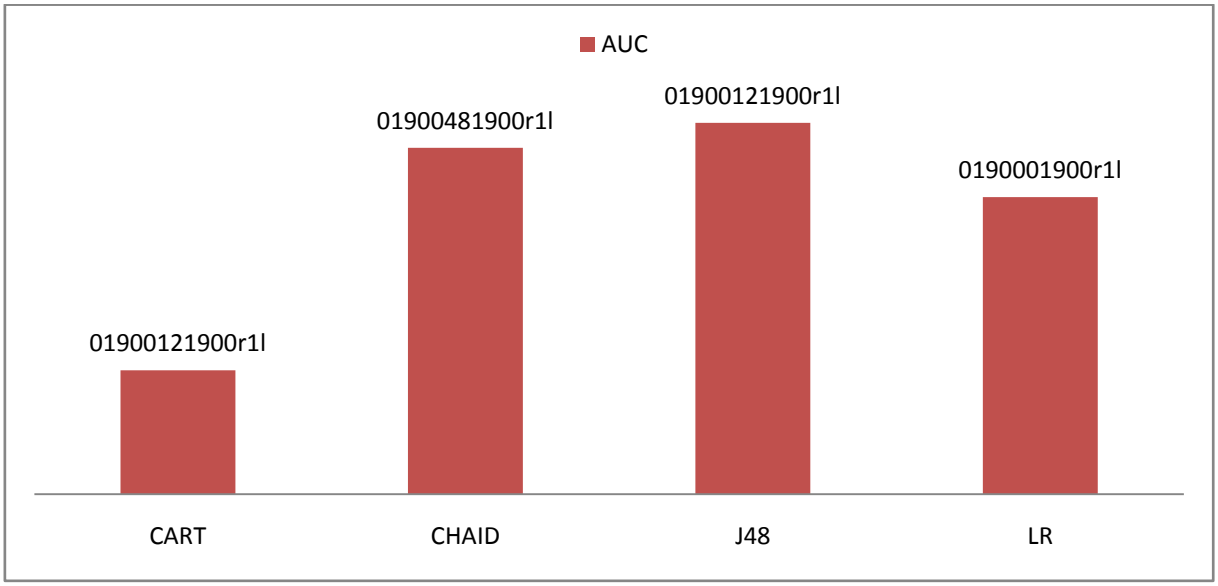
Tablo 10. Bağımsız değişkenlerin tümü kategorik olduğu durumda yöntemlere göre sınıflandırma sonuçları (1000 deneme).

		CART	CHAID	J48	LR
1000 deneme	Duyarlılık (%)	79,9	85,0	85,9	82,5
	Özgüllük (%)	80,0	80,0	86,6	83,5
	PKD (%)	80,1	84,2	86,5	83,4
	NKD (%)	80,0	84,9	86,0	83,1
	Doğruluk (%)	80,0	84,5	86,3	83,2
	AUC	0,83	0,92	0,93	0,90
	AUC'nin Standart Hatası	0,009	0,006	0,007	0,006



Sekil 7. Bağımsız değişkenlerin tümü kategorik olduğu durumda yöntemlere göre duyarlılık değerleri (1000 deneme)

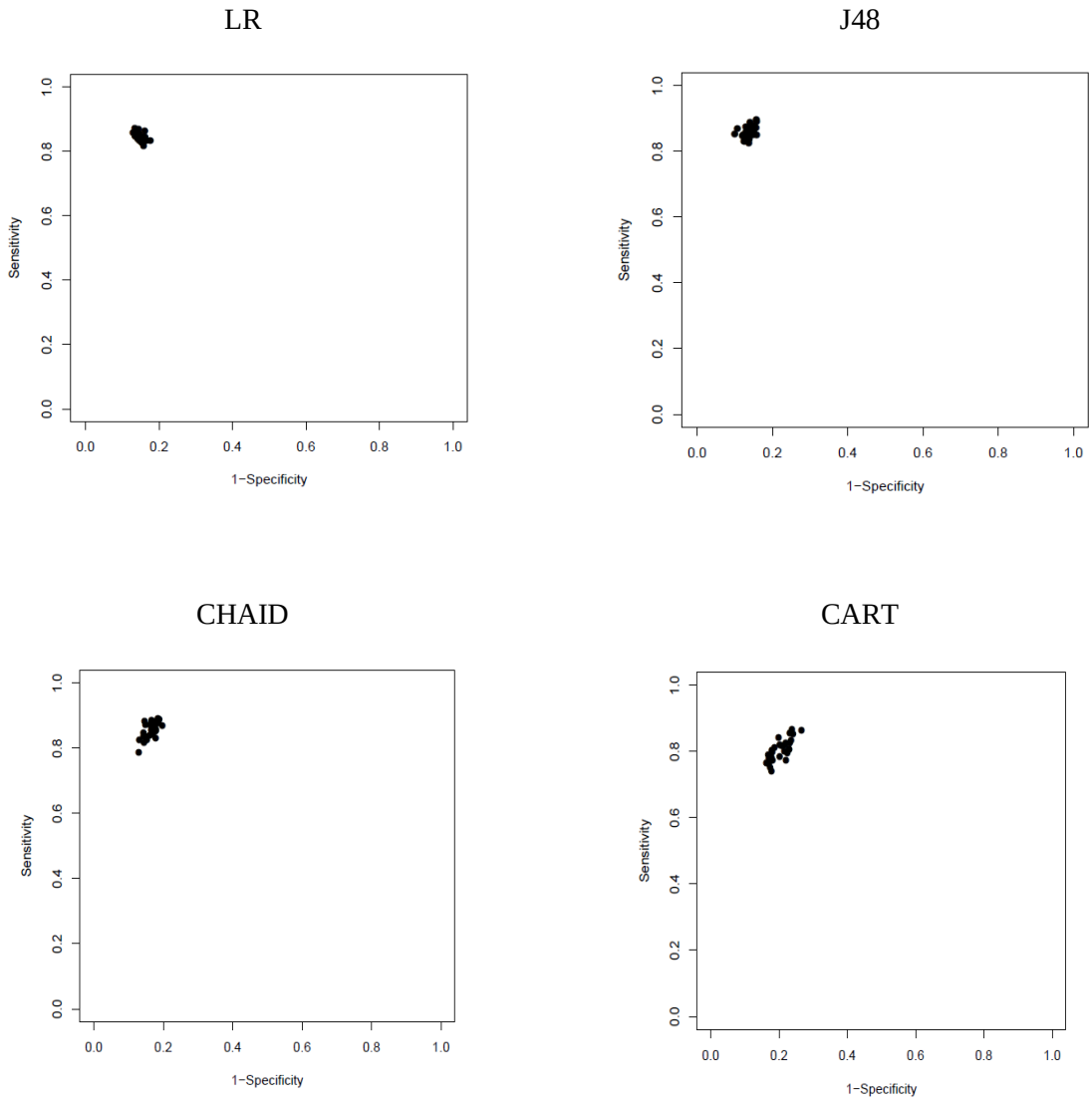
Bağımsız değişkenlerin tümünün kategorik olduğu 1000 denemelik çalışma sonuçlarında görüldüğü gibi en yüksek duyarlılık oranı J48 yönteminde (%85,9), en düşük duyarlılık oranı CART yönteminde (%79,9) gözlenmiştir (Şekil 7).



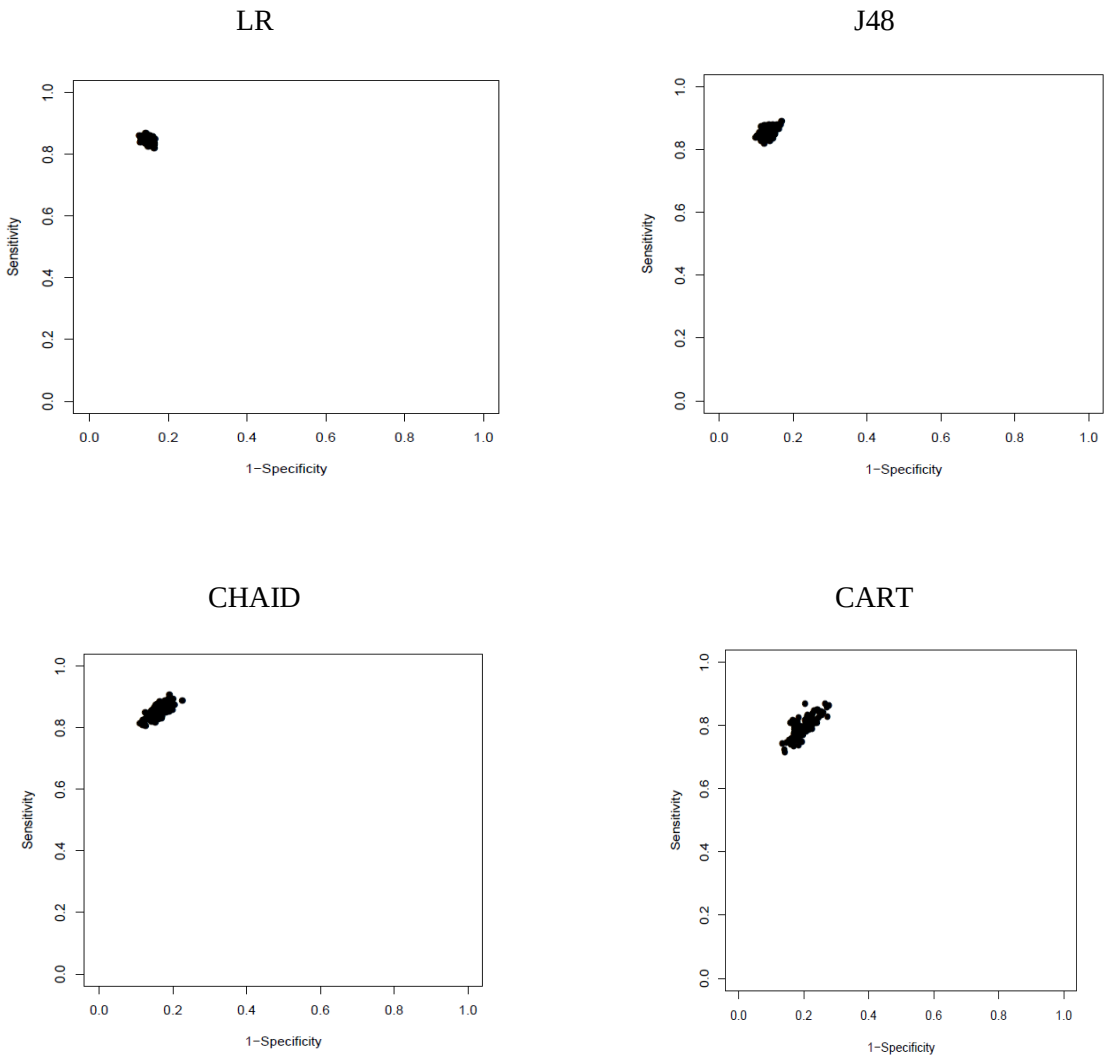
Sekil 8. Bağımsız değişkenlerin tümü kategorik olduğu durumda yöntemlere göre eğri altında kalan alan (AUC) değerleri (1000 deneme)

Bağımsız değişkenlerin tümünün kategorik olduğu 1000 denemelik çalışma sonucunda grafikte de görüldüğü gibi en yüksek AUC değeri J48 yönteminde (0,93), en düşük AUC değeri CART yönteminde (0,83) gözlenmiştir (Şekil 8).

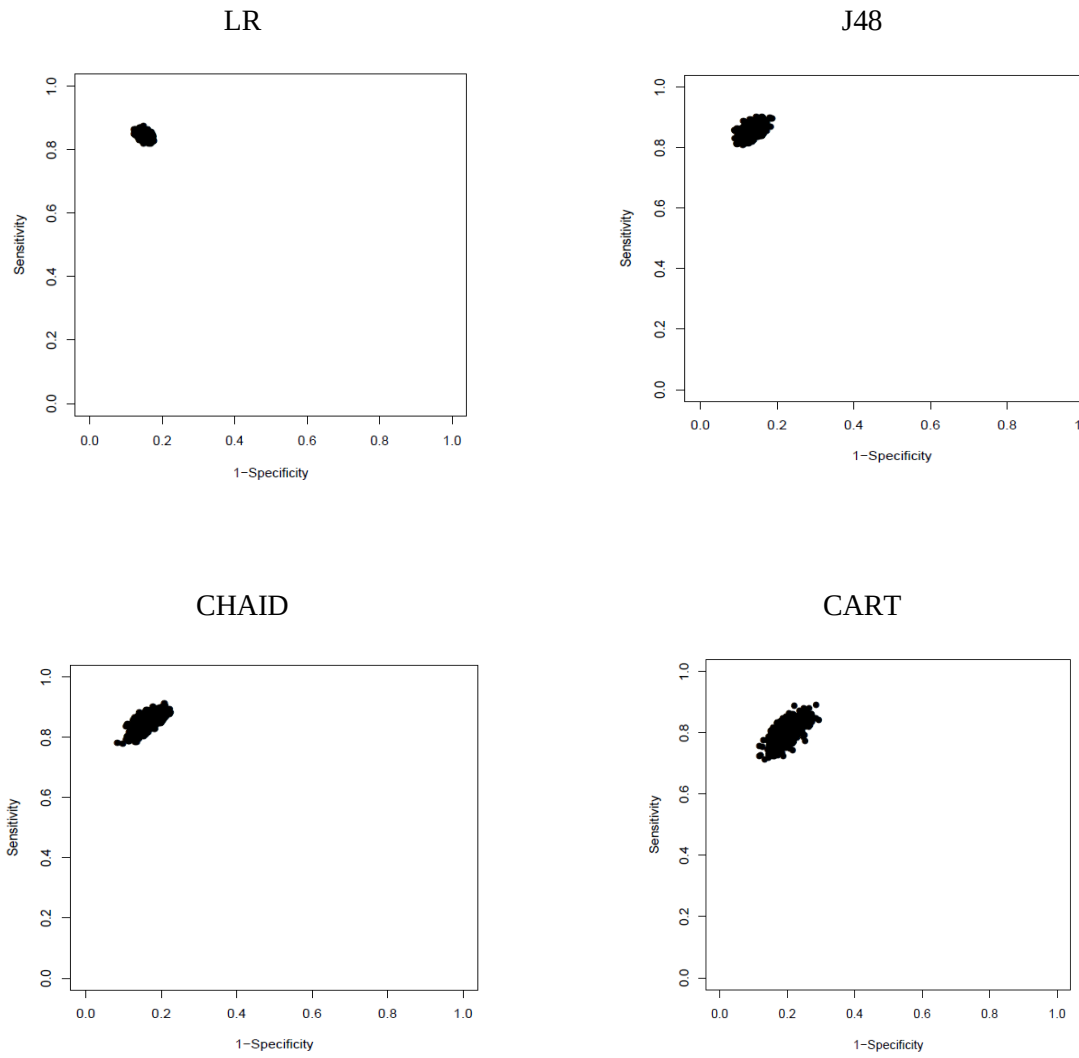
Tümü kategorik yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-Spesifite değerlerinin grafiksel gösterimi (Şekil 9-11)'de gösterildi (30, 100, 1000 deneme).



Şekil 9. Tümü kategorik yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-spesifite değerlerinin grafiksel gösterimi (30 deneme)



Şekil 10. Tümü kategorik yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-spesifite değerlerinin grafiksel gösterimi (100 deneme)



Şekil 11. Tümü kategorik yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-spesifite değerlerinin grafiksel gösterimi (1000 deneme)

5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 30 denemelik simülasyon çalışması sonuçlarına ilişkin duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC oranları (Tablo 11)'de gösterildi. Bu sonuçlara göre, normal (N) dağılımda dört yöntem arasında en düşük doğruluk oranı CART yönteminde (%80,1) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%90,9) gözlenmiştir. F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%77,7) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%91,8) gözlenmiştir. N-F dağılımında dört yöntem arasında en düşük doğruluk oranı LR ve CART yöntemlerinde (%80,6) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%91,5) gözlenmiştir.

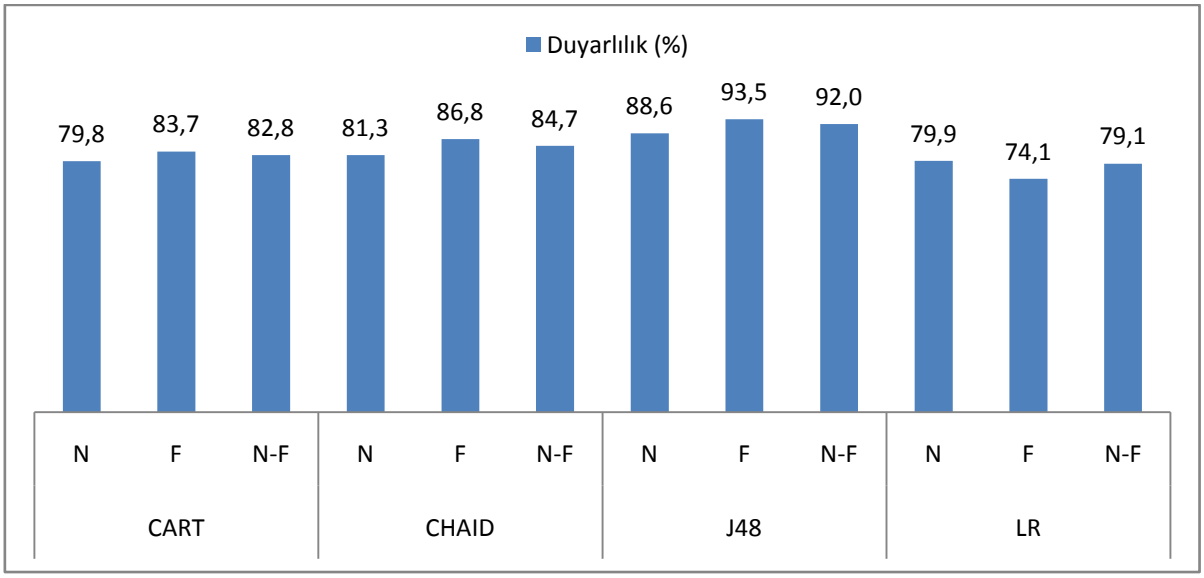
Tablo 11. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre sınıflandırma sonuçları (30 deneme).

		Algoritmalar											
		CART			CHAID			J48			LR		
		N	F	N-F	N	F	N-F	N	F	N-F	N	F	N-F
30 deneme	Duyarlılık (%)	79,8	83,7	82,8	81,3	86,8	84,7	88,6	93,5	92,0	79,9	74,1	79,1
	Özgüllük (%)	80,3	81,1	78,5	83,7	83,0	82,6	93,1	90,2	91,0	82,4	81,3	82,0
	PKD (%)	80,4	81,7	79,5	83,4	83,7	83,1	92,8	90,5	91,1	81,9	79,9	81,5
	NKD (%)	80,1	83,4	82,2	81,8	86,3	84,5	89,1	93,2	92,0	80,4	75,9	79,7
	Doğruluk (%)	80,1	82,4	80,6	82,5	84,9	83,7	90,9	91,8	91,5	81,1	77,7	80,6
	AUC	0,83	0,85	0,83	0,91	0,93	0,92	0,94	0,95	0,95	0,89	0,86	0,88
	AUC'nin Standart Hatası	0,009	0,008	0,009	0,006	0,006	0,006	0,005	0,005	0,005	0,007	0,008	0,007

N: 5 sürekli değişken normal dağılımdan türetilmiştir.

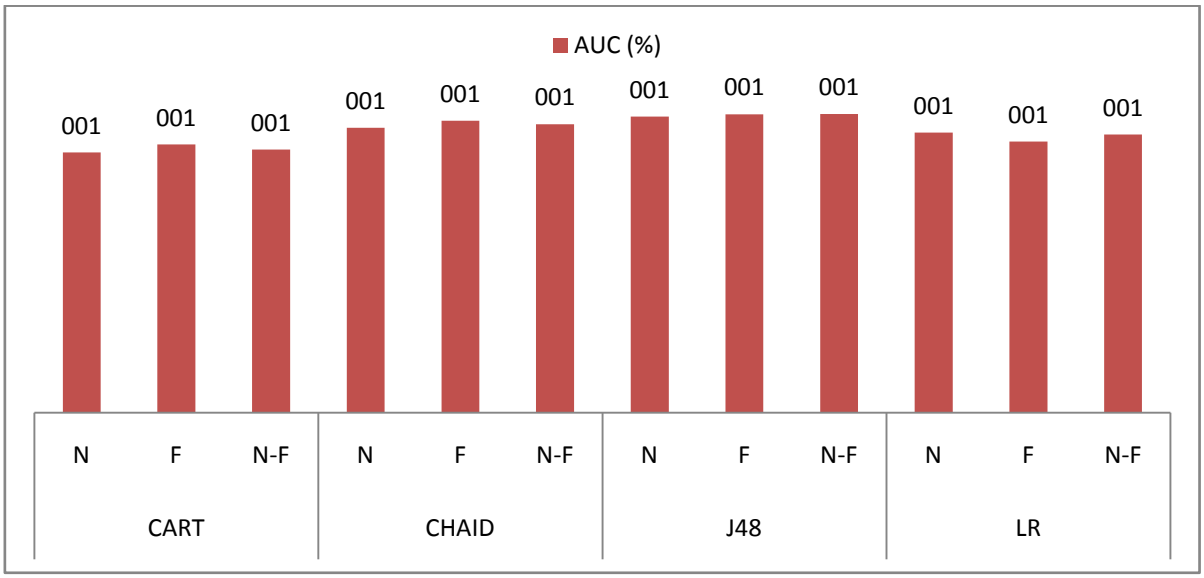
F: 5 sürekli değişken F dağılımından türetilmiştir.

N-F: 3 değişken F dağılımından, 2 değişken normal dağılımdan türetilmiştir.



Sekil 12. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre duyarlılık değerleri (30 deneme)

5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 30 denemelik simülasyon çalışması sonuçlarında duyarlılık grafiğinde de görüldüğü gibi en yüksek duyarlılık oranı (%93,5) F dağılımında J48 yönteminde gözlenmiştir (Şekil 12).



Sekil 13. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre eğri altında kalan alan (AUC) değerleri (30 deneme)

5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 30 denemelik simülasyon çalışması sonuçlarında en yüksek AUC oranı (0,95) F ve N-F dağılımlarında J48 yönteminde gözlenmiştir (Şekil 13).

5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 100 denemelik simülasyon çalışması sonuçlarına ilişkin duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC oranları (Tablo 12)'de gösterildi. Bu sonuçlara göre, normal (N) dağılımda dört yöntem arasında en düşük doğruluk oranı CART yönteminde (%79,9) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%91,0) gözlenmiştir. F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%78,0) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%92,0) gözlenmiştir. N-F dağılımında dört yöntem arasında en düşük doğruluk oranı LR ve CART yöntemlerinde (%80,8) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%91,4) gözlenmiştir.

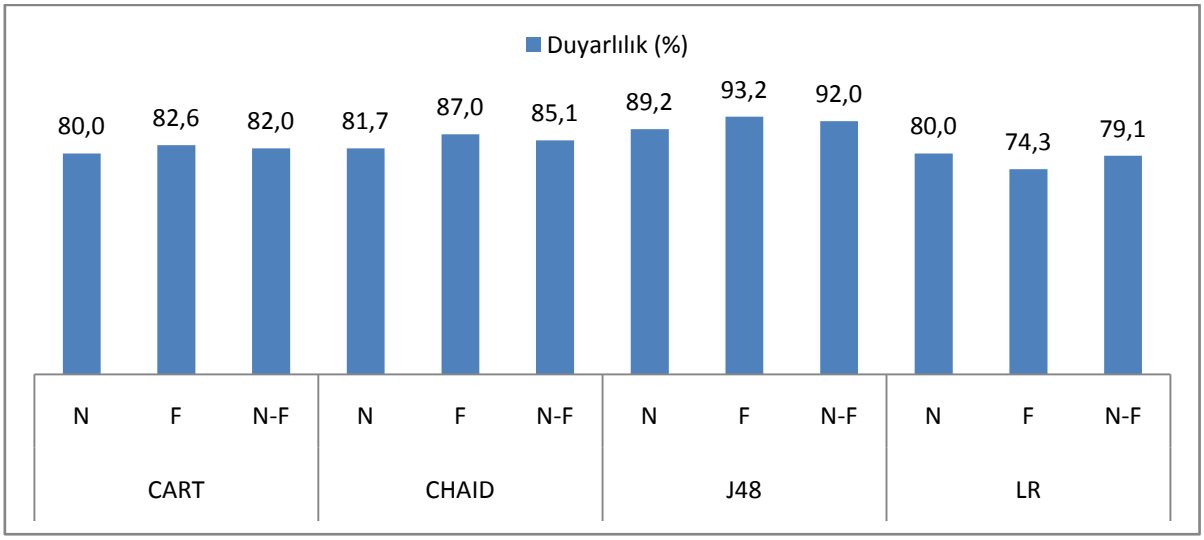
Tablo 12. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre sınıflandırma sonuçları (100 deneme).

		Algoritmalar											
		CART			CHAID			J48			LR		
		N	F	N-F	N	F	N-F	N	F	N-F	N	F	N-F
100 deneme	Duyarlılık (%)	80,0	82,6	82,0	81,7	87,0	85,1	89,2	93,2	92,0	80,0	74,3	79,2
	Özgüllük (%)	79,9	82,3	79,5	84,3	83,9	82,7	92,9	90,7	90,8	82,5	81,6	82,5
	PKD (%)	80,1	82,5	80,2	83,9	84,4	83,1	92,7	91,0	90,9	82,1	80,2	81,9
	NKD (%)	80,1	82,7	81,7	82,3	86,6	84,8	89,6	93,1	91,9	80,5	76,1	79,8
	Doğruluk (%)	79,9	82,5	80,8	83,0	85,4	83,9	91,0	92,0	91,4	81,3	78,0	80,8
	AUC	0,83	0,85	0,84	0,91	0,93	0,92	0,95	0,95	0,95	0,89	0,87	0,89
	AUC'nin Standart Hatası	0,009	0,008	0,009	0,006	0,005	0,006	0,005	0,005	0,005	0,007	0,008	0,007

N: 5 sürekli değişken normal dağılımdan türetilmiştir.

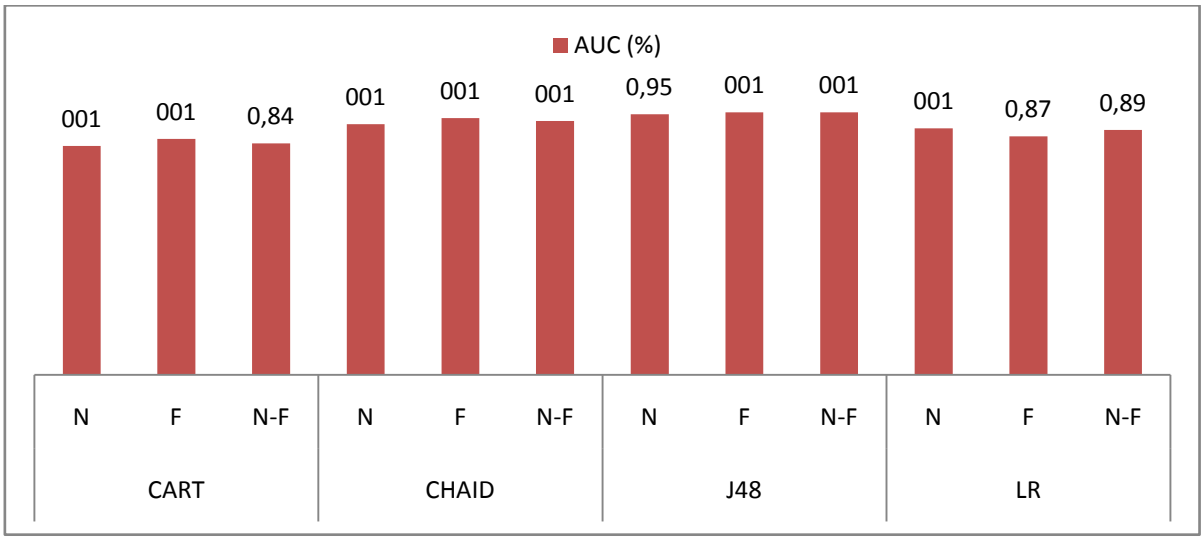
F: 5 sürekli değişken F dağılımından türetilmiştir

N-F: 3 değişken F dağılımından, 2 değişken normal dağılımdan türetilmiştir



Sekil 14. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre duyarlılık değerleri (100 deneme)

5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 100 denemelik simülasyon çalışması sonuçlarında en yüksek duyarlılık oranı F dağılımında J48 yönteminde gözlenirken en düşük duyarlılık oranı yine F dağılımında LR yönteminde gözlenmiştir (Şekil 14).



Şekil 15. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre eğri altında kalan alan (AUC) değerleri (100 deneme)

5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 100 denemelik çalışma sonuçlarında en yüksek AUC oranı N, F ve N-F dağılımlarında J48 yönteminde (0,95) gözlenirken en düşük AUC değeri N dağılımında CART yönteminde (0,83) gözlenmiştir (Şekil 15).

5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 1000 denemelik simülasyon çalışması sonuçlarına ilişkin duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC oranları (Tablo 13)'de gösterildi. Bu sonuçlara göre, normal (N) dağılımda dört yöntem arasında en düşük doğruluk oranı CART yönteminde (%80,0) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%90,8) gözlenmiştir. F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%78,0) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%91,9) gözlenmiştir. N-F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%80,9) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%91,4) gözlenmiştir.

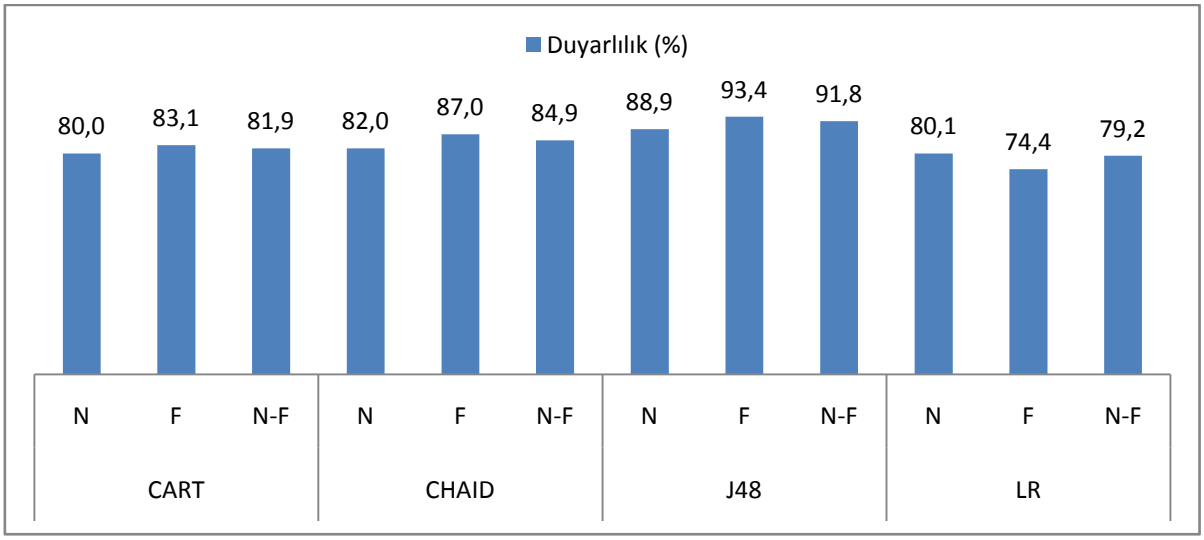
Tablo 13. 5 kategorik 5 sürekli bağımsız değişkenlerin 1000 denemelik simülasyon çalışması sonuçları.

		Algoritmalar											
		CART			CHAID			J48			LR		
		N	F	N-F	N	F	N-F	N	F	N-F	N	F	N-F
1000 deneme	Duyarlılık (%)	80,0	83,1	81,9	82,0	87,0	84,9	88,9	93,4	91,8	80,1	74,4	79,2
	Özgüllük (%)	80,0	81,9	80,0	83,6	83,6	83,2	92,8	90,4	91,1	82,6	81,5	82,6
	PKD (%)	80,2	82,3	80,5	83,4	84,2	83,6	92,5	90,7	91,2	82,1	80,1	82,0
	NKD (%)	80,1	83,1	81,7	82,3	86,6	84,7	89,3	93,2	91,8	80,6	76,1	79,9
	Doğruluk (%)	80,0	82,5	81,0	82,8	85,3	84,0	90,8	91,9	91,4	81,3	78,0	80,9
	AUC	0,83	0,85	0,84	0,91	0,93	0,92	0,94	0,95	0,95	0,89	0,86	0,89
	AUC'nin Standart Hatası	0,009	0,008	0,009	0,006	0,005	0,006	0,005	0,005	0,005	0,007	0,008	0,007

N: 5 sürekli değişken normal dağılımdan türetilmiştir

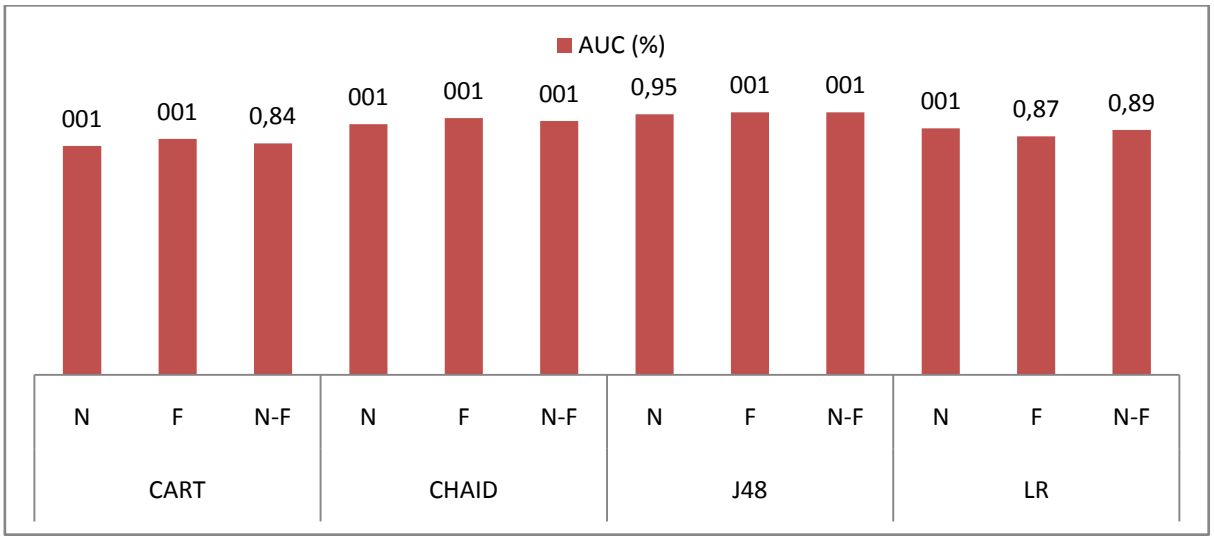
F: 5 sürekli değişken F dağılımından türetilmiştir

N-F: 3 değişken F dağılımından, 2 değişken normal dağılımdan türetilmiştir



Sekil 16. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 1000 denemelik simülasyon çalışması sonuçlarında duyarlılık

5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 1000 denemelik çalışma sonuçlarında en yüksek duyarlılık oranı F dağılımında J48 yönteminde gözlenirken (%93,4) en düşük duyarlılık oranı yine F dağılımında LR yönteminde gözlenmiştir (Şekil 16).

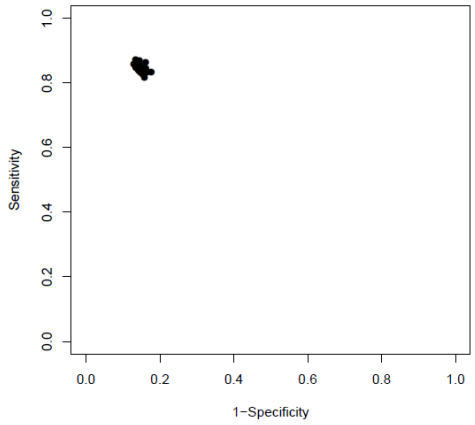


Şekil 17. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre eğri altında kalan alan (AUC) değerleri (1000 deneme)

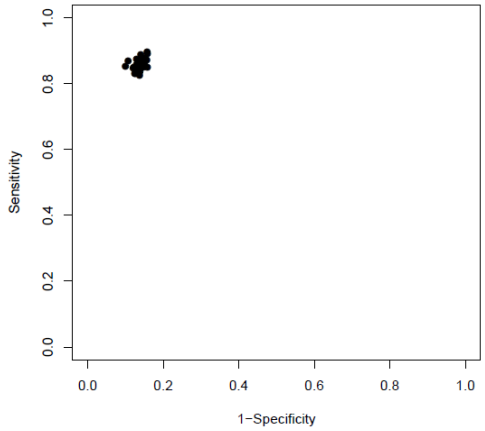
5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 1000 denemelik çalışma sonuçlarında en yüksek AUC oranı N, F ve N-F dağılımlarında J48 yönteminde (0,95) gözlenirken en düşük AUC değeri N dağılımında CART yönteminde (0,83) gözlenmiştir (Şekil 17).

5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için algoritmalara gore Sensitivity değerlerine karşılık 1-Spesifite değerlerinin grafiksel gösterimi (Şekil 18-20)'de gösterildi (30, 100, 1000 deneme).

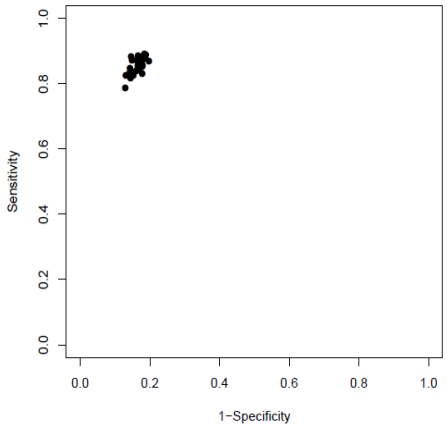
LR



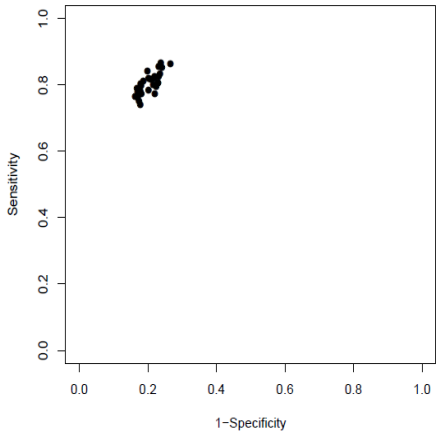
J48



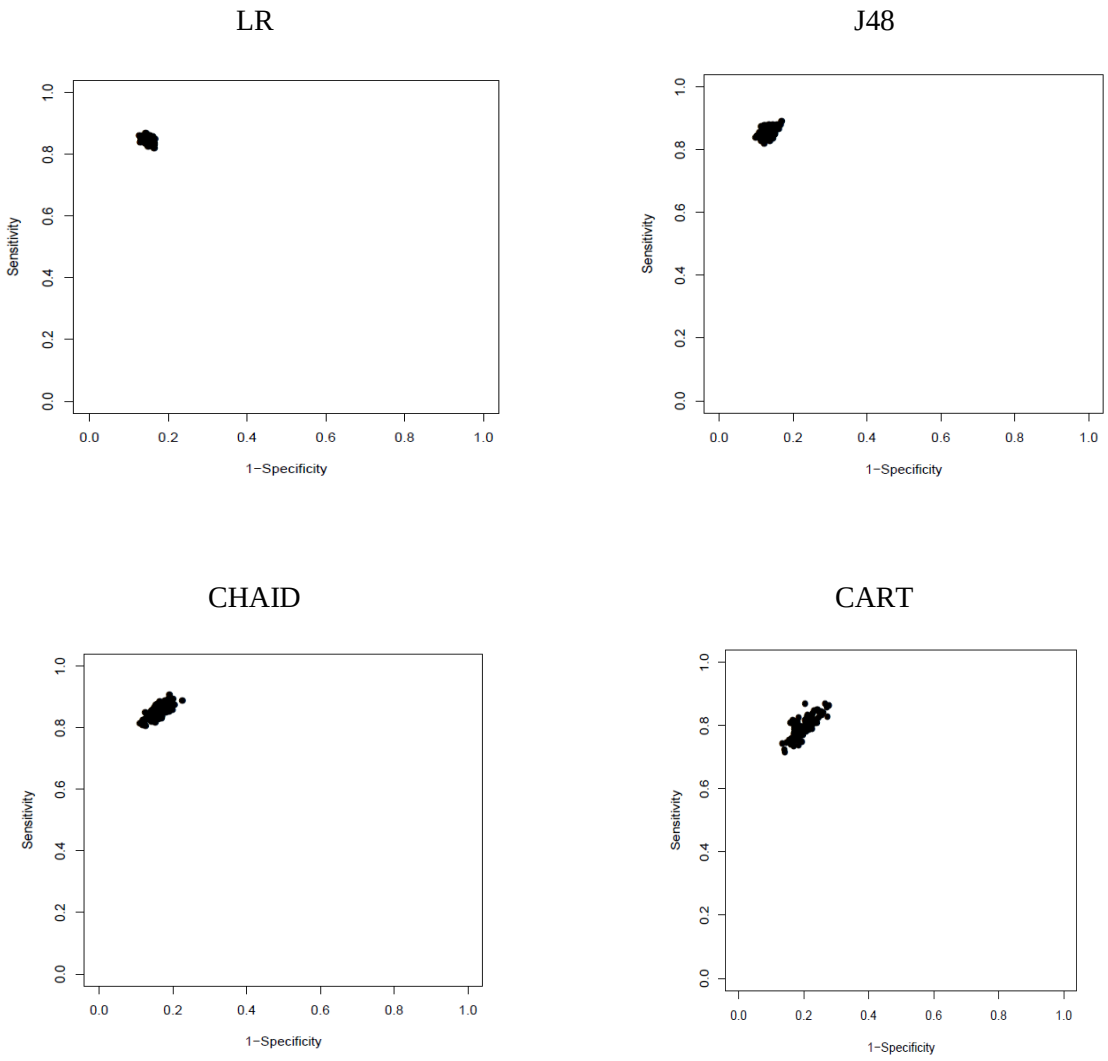
CHAID



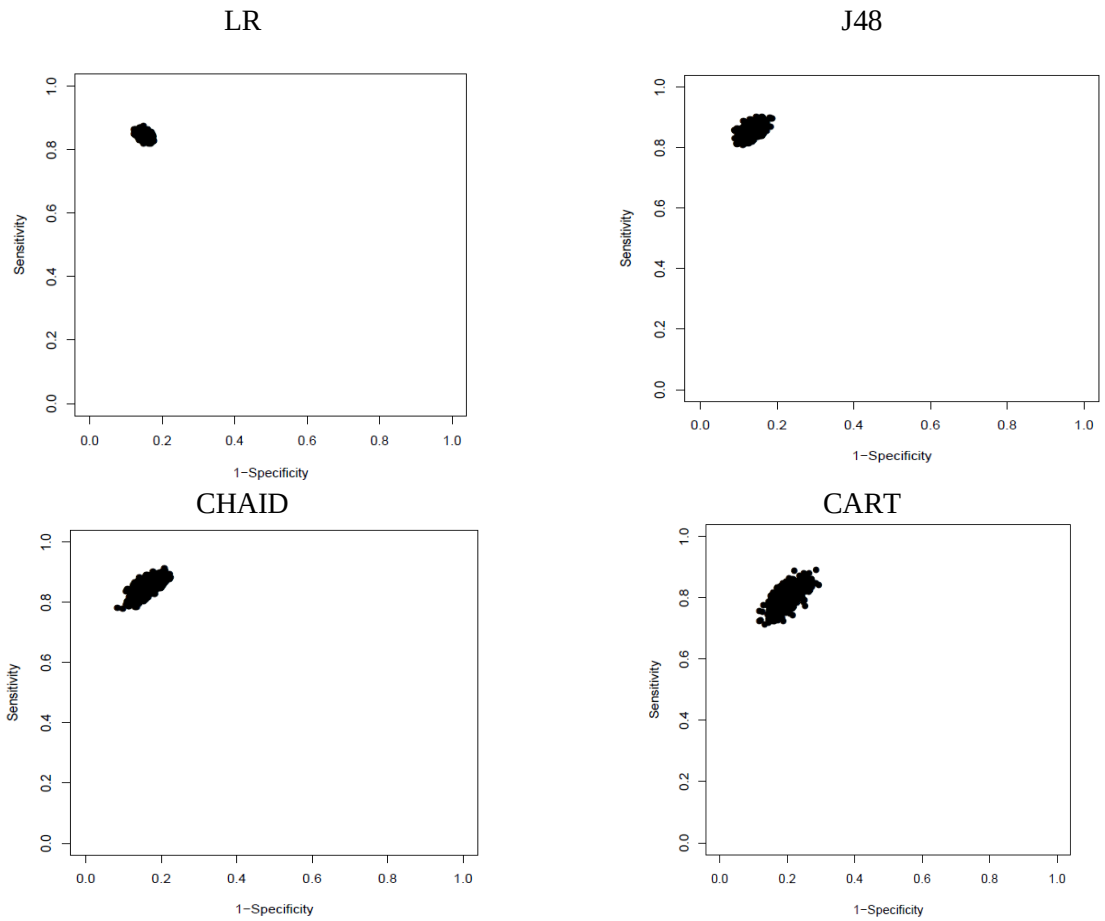
CART



Şekil 18. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-spesifite değerlerinin grafiksel gösterimi (30 deneme)



Şekil 19. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-spesifite değerlerinin grafiksel gösterimi (100 deneme)



Şekil 20. 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-spesifite değerlerinin grafiksel gösterimi (1000 deneme)

Tümü Sürekli yapıda olan bağımsız değişkenler için 30 denemelik simülasyon çalışması sonuçlarına ilişkin duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC oranları (Tablo 14)'de gösterildi. Bu sonuçlara göre, normal (N) dağılımda dört yöntem arasında en düşük doğruluk oranı CART yönteminde (%81,0) gözlenirken en yüksek doğruluk oranı J48 yönteminde (%92,3) gözlenmiştir. F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%73,2) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%89,7) gözlenmiştir. N-F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%76,2) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%91,2) gözlenmiştir.

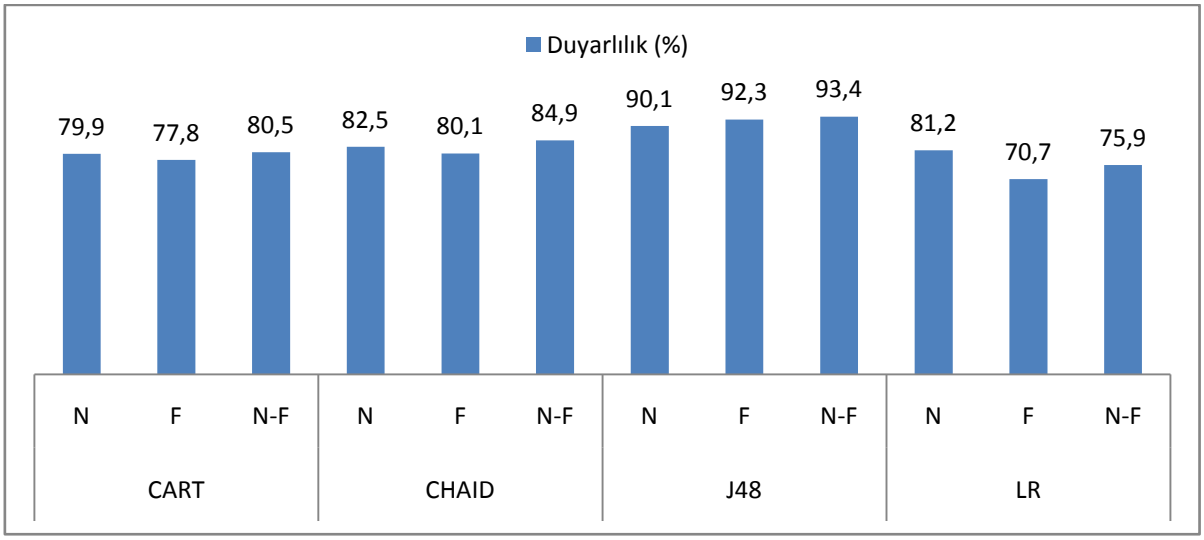
Tablo 14. Tümü sürekli yapıda bağımsız değişkenlerin 30 denemelik simülasyon çalışması sonuçları.

		Algoritmalar											
		CART			CHAID			J48			LR		
		N	F	N-F	N	F	N-F	N	F	N-F	N	F	N-F
30 deneme	Duyarlılık (%)	79,9	77,8	80,5	82,5	80,1	84,9	90,1	92,3	93,4	81,2	70,7	75,9
	Özgüllük (%)	82,0	74,2	78,2	83,2	78,8	79,4	94,4	87,1	89,1	83,9	75,7	76,5
	PKD (%)	81,7	75,2	78,8	83,3	79,2	80,6	94,2	87,8	89,6	83,5	74,4	76,4
	NKD (%)	80,4	77,1	80,2	82,9	79,9	84,1	90,6	91,9	93,1	81,7	72,1	76,0
	Doğruluk (%)	81,0	76,0	79,3	82,8	79,4	82,1	92,3	89,7	91,2	82,6	73,2	76,2
	AUC	0,84	0,79	0,82	0,91	0,87	0,90	0,96	0,94	0,96	0,90	0,81	0,84
	AUC'nin Standart Hatası	0,009	0,010	0,009	0,006	0,008	0,007	0,004	0,005	0,004	0,007	0,010	0,009

N: 5 sürekli değişken normal dağılımdan türetilmiştir.

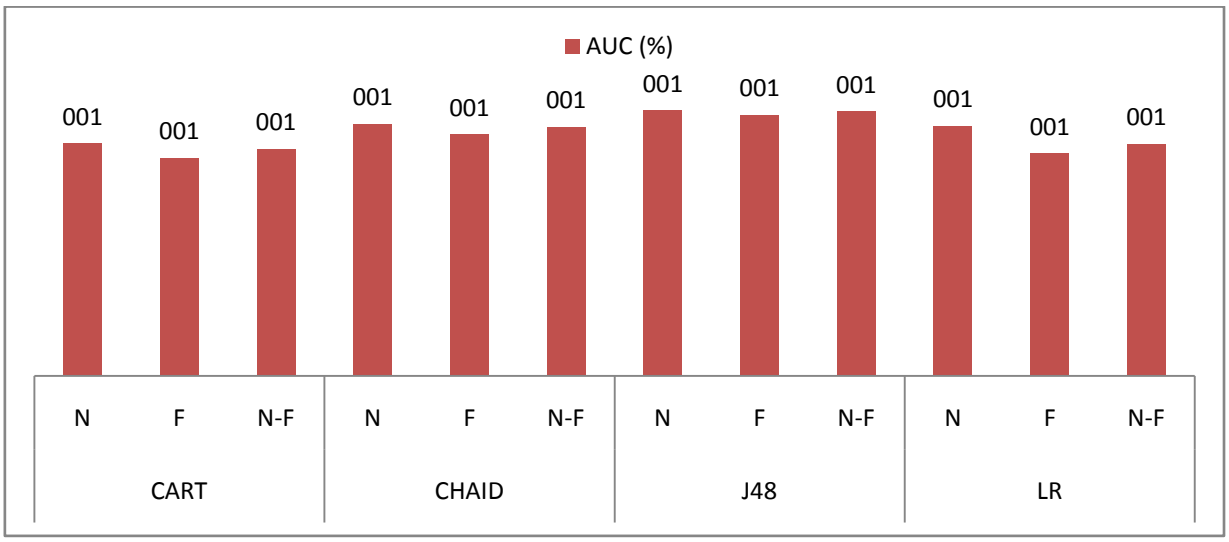
F: 5 sürekli değişken F dağılımından türetilmiştir.

F: 3 değişken F dağılımından, 2 değişken normal dağılımdan türetilmiştir.



Şekil 21. Tümü sürekli yapıda bağımsız değişkenlerin 30 denemelik simülasyon çalışması sonuçlarında duyarlılık

Tümü Sürekli yapıda bağımsız değişkenlerin 30 denemelik simülasyon çalışması sonuçlarında en yüksek duyarlılık oranı N-F dağılımında J48 yönteminde gözlenirken (%93,4) en düşük duyarlılık oranı F dağılımında LR yönteminde gözlenmiştir (Şekil 21).



Şekil 22. Tümü sürekli yapıda bağımsız değişkenlerin 30 denemelik simülasyon çalışması sonuçlarında AUC

Tümü sürekli yapıda bağımsız değişkenlerin 30 denemelik simülasyon çalışması sonuçlarında en yüksek AUC oranı N ve N-F dağılımlarında J48 yönteminde (0,96) gözlenirken en düşük AUC değeri F dağılımında CART yönteminde (0,79) gözlenmiştir (Şekil 22).

Sürekli yapıda olan bağımsız değişkenler için 100 denemelik simülasyon çalışması sonuçlarına ilişkin duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC oranları (Tablo 15)'de gösterildi. Bu sonuçlara göre, normal (N) dağılımda dört yöntem arasında en düşük doğruluk oranı CART yönteminde (%81,1) gözlenirken en yüksek doğruluk oranı J48 yönteminde (%91,3) gözlenmiştir. F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%73,2) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%89,5) gözlenmiştir. N-F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%76,2) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%91,3) gözlenmiştir.

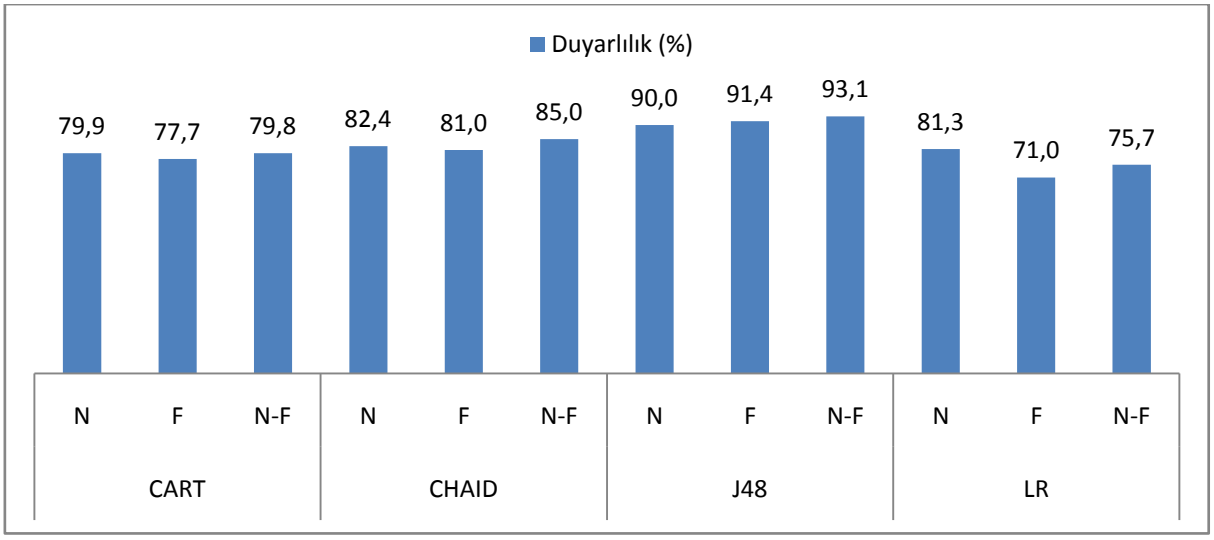
Tablo 15. Tümü sürekli yapıda bağımsız değişkenlerin 100 denemelik simülasyon çalışması sonuçları.

		Algoritmalar											
		CART			CHAID			J48			LR		
		N	F	N-F	N	F	N-F	N	F	N-F	N	F	N-F
100 deneme	Duyarlılık (%)	79,9	77,7	79,8	82,4	81,0	85,0	90,0	91,4	93,1	81,3	71,0	75,7
	Özgüllük (%)	82,4	74,5	78,5	83,5	77,5	79,1	94,0	87,6	89,4	83,7	75,4	76,8
	PKD (%)	82,0	75,4	78,9	83,5	78,4	80,3	89,9	88,1	89,9	83,3	74,3	76,5
	NKD (%)	80,4	77,1	79,7	82,8	80,5	84,1	92,9	91,1	92,9	81,8	72,2	75,9
	Doğruluk (%)	81,1	76,1	79,3	83,0	79,3	82,0	91,3	89,5	91,3	82,5	73,2	76,2
	AUC	0,84	0,79	0,82	0,91	0,87	0,90	0,96	0,94	0,96	0,90	0,81	0,84
	AUC'nin Standart Hatası	0,009	0,010	0,009	0,006	0,008	0,007	0,004	0,005	0,004	0,007	0,010	0,009

N: 5 sürekli değişken normal dağılımdan türetilmiştir.

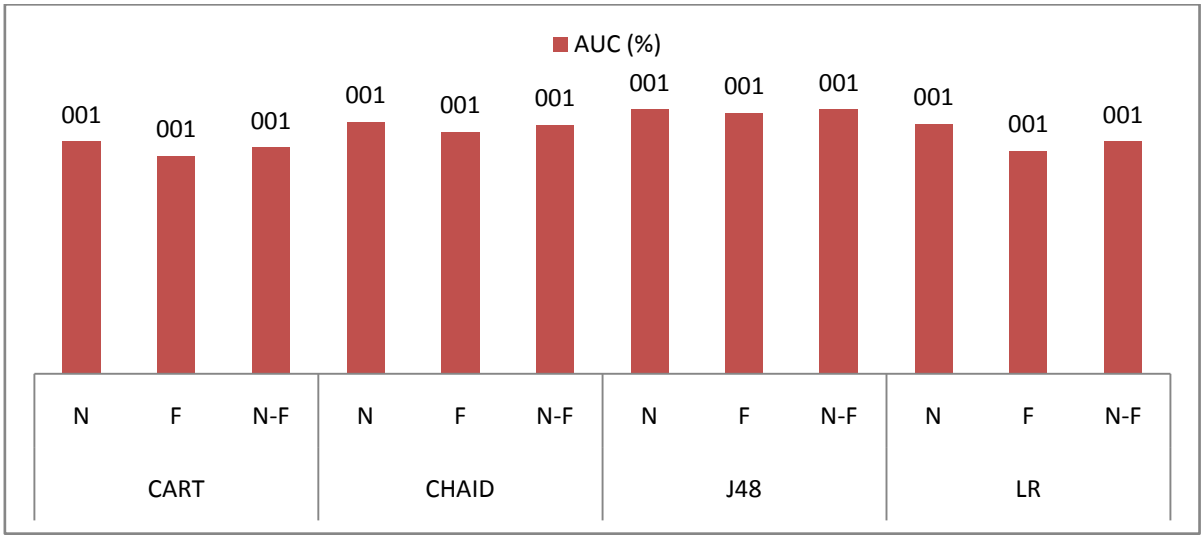
F: 5 sürekli değişken F dağılımından türetilmiştir.

N-F: 3 değişken F dağılımından, 2 değişken normal dağılımdan türetilmiştir.



Şekil 23. Tümü sürekli yapıda bağımsız değişkenlerin 100 denemelik simülasyon çalışması sonuçlarında duyarlılık.

Tümü sürekli yapıda bağımsız değişkenlerin 100 denemelik simülasyon çalışması sonuçlarında en yüksek duyarlılık oranı N-F dağılımında J48 yönteminde gözlenirken (%93,1) en düşük duyarlılık oranı F dağılımında LR yönteminde (%71,0) gözlenmiştir (Şekil 23).



Şekil 24. Tümü sürekli yapıda bağımsız değişkenlerin 100 denemelik simülasyon çalışması sonuçlarında AUC.

Sürekli bağımsız değişkenlerin tümü kategorik 100 denemelik çalışma sonuçlarında en yüksek AUC oranı N ve N-F dağılımlarında J48 yönteminde (0,96) gözlenirken en düşük AUC değeri F dağılımında CART yönteminde (0,79) gözlenmiştir (Şekil 24).

Sürekli yapıda olan bağımsız değişkenler için 1000 denemelik simülasyon çalışması sonuçlarına ilişkin duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC oranları (Tablo 16)'da gösterildi. Bu sonuçlara göre, normal (N) dağılımda dört yöntem arasında en düşük doğruluk oranı CART yönteminde (%80,8) gözlenirken en yüksek doğruluk oranı J48 yönteminde (%91,9) gözlenmiştir. F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%73,3) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%89,4) gözlenmiştir. N-F dağılımında dört yöntem arasında en düşük doğruluk oranı LR yönteminde (%76,1) gözlenirken en yüksek doğruluk oranı J48 algoritmasında (%91,4) gözlenmiştir.

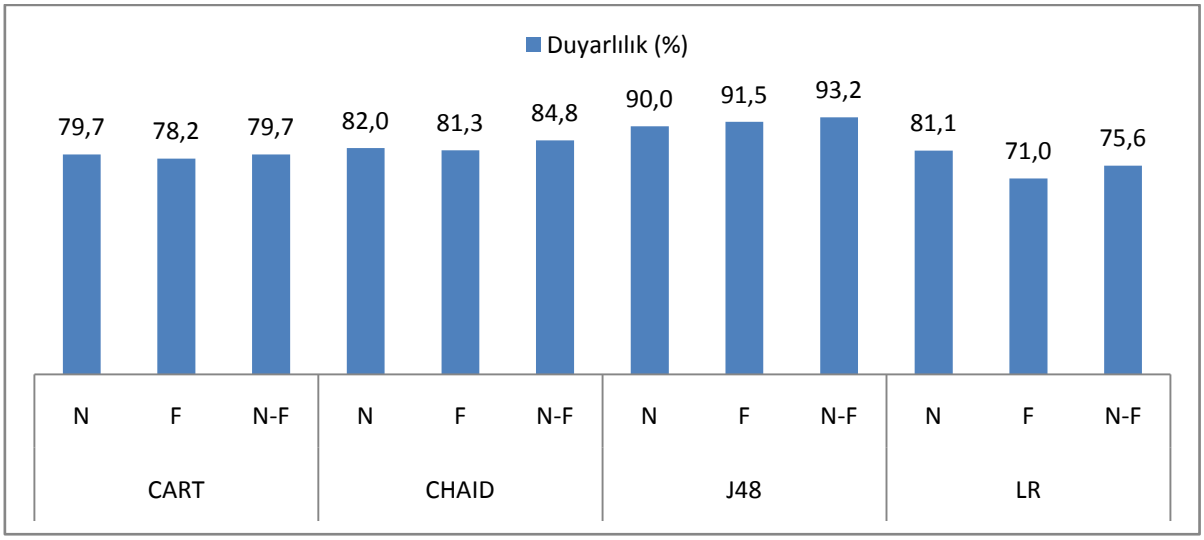
Tablo 16. Tümü sürekli yapıda bağımsız değişkenlerin 1000 denemelik simülasyon çalışması sonuçları.

		Algoritmalar											
		CART			CHAID			J48			LR		
		N	F	N-F	N	F	N-F	N	F	N-F	N	F	N-F
1000 deneme	Duyarlılık (%)	79,7	78,2	79,7	82,0	81,3	84,8	90,0	91,5	93,2	81,1	71,0	75,6
	Özgüllük (%)	82,0	74,2	78,4	83,7	77,2	79,2	93,9	87,3	89,6	83,7	75,7	76,6
	PKD (%)	81,7	75,3	78,7	83,6	78,2	84,8	93,7	87,9	90,0	83,3	74,5	76,4
	NKD (%)	80,2	77,4	79,5	82,4	80,6	79,2	90,4	91,2	93,0	81,6	72,3	75,9
	Doğruluk (%)	80,8	76,2	79,0	82,9	79,2	82,0	91,9	89,4	91,4	82,4	73,3	76,1
	AUC	0,84	0,79	0,82	0,91	0,87	0,90	0,96	0,94	0,96	0,90	0,81	0,84
	AUC'nin Standart Hatası	0,009	0,010	0,009	0,006	0,008	0,007	0,004	0,005	0,004	0,007	0,010	0,009

N: 5 sürekli değişken normal dağılımdan türetilmiştir.

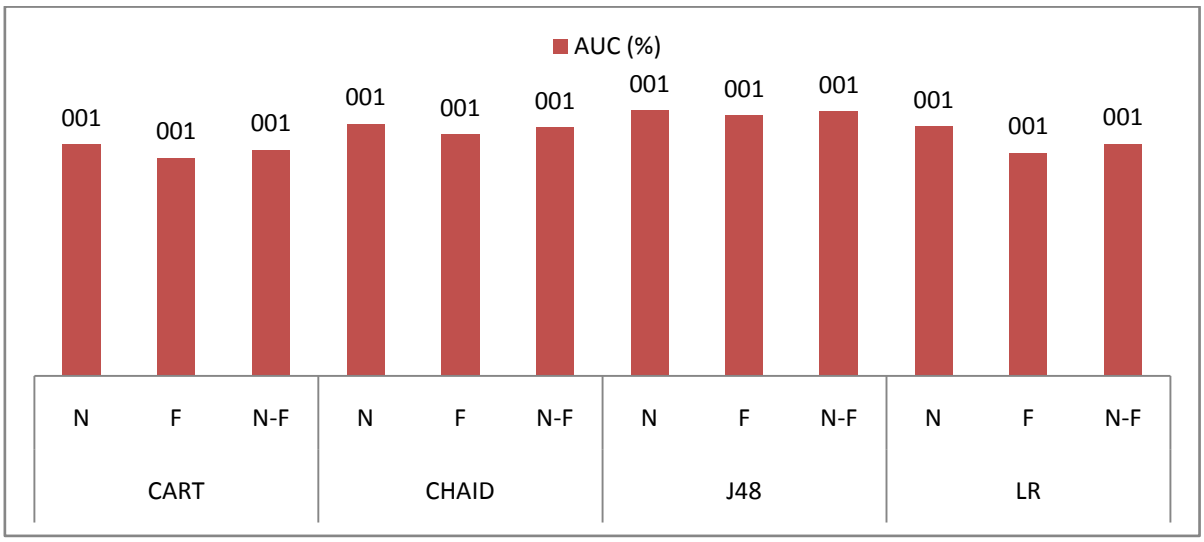
F: 5 sürekli değişken F dağılımından türetilmiştir.

N-F: 3 değişken F dağılımından, 2 değişken normal dağılımdan türetilmiştir.



Şekil 25. Tümü sürekli yapıda bağımsız değişkenlerin 1000 denemelik simülasyon çalışması sonuçlarında duyarlılık

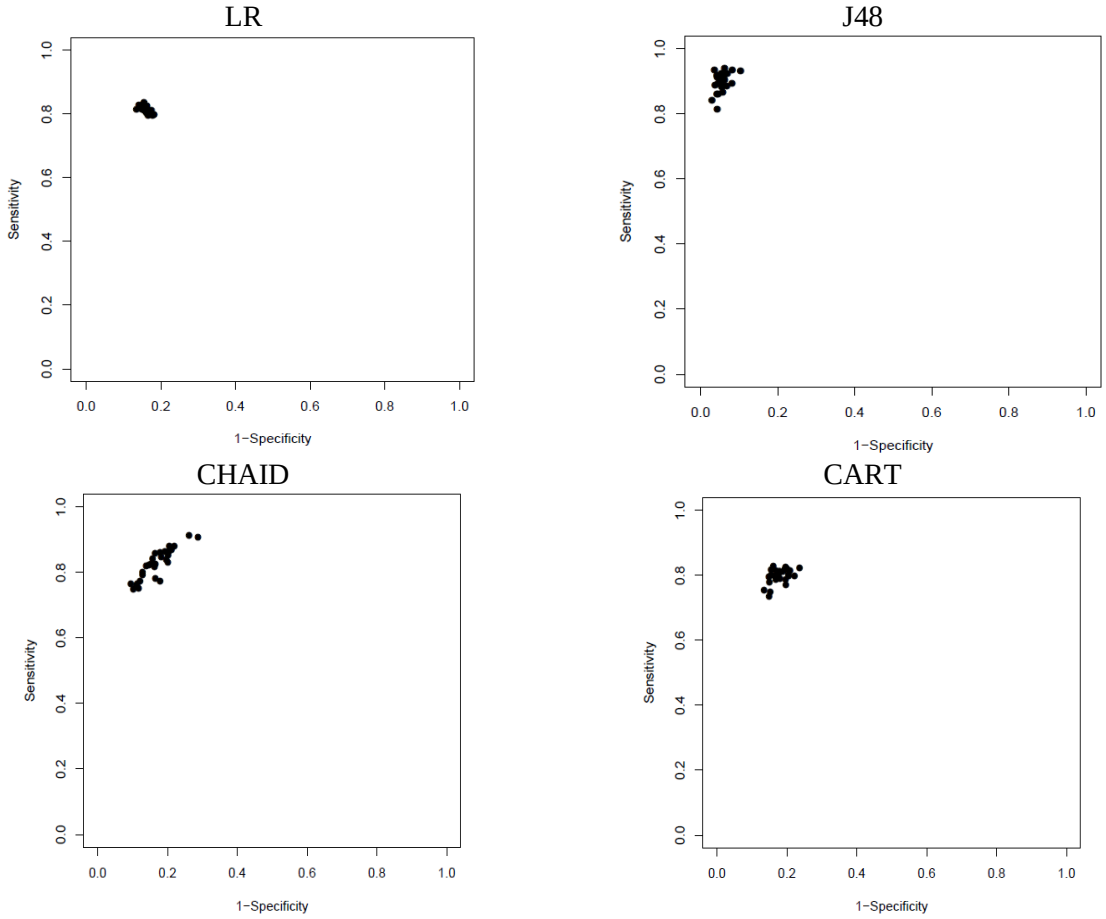
Tümü Sürekli yapıda bağımsız değişkenlerin 1000 denemelik çalışma sonuçlarında en yüksek duyarlılık oranı N-F dağılımında J48 yönteminde (%93,2) gözlenirken en düşük duyarlılık oranı F dağılımında LR yönteminde (%71,0) gözlenmiştir (Şekil 25).



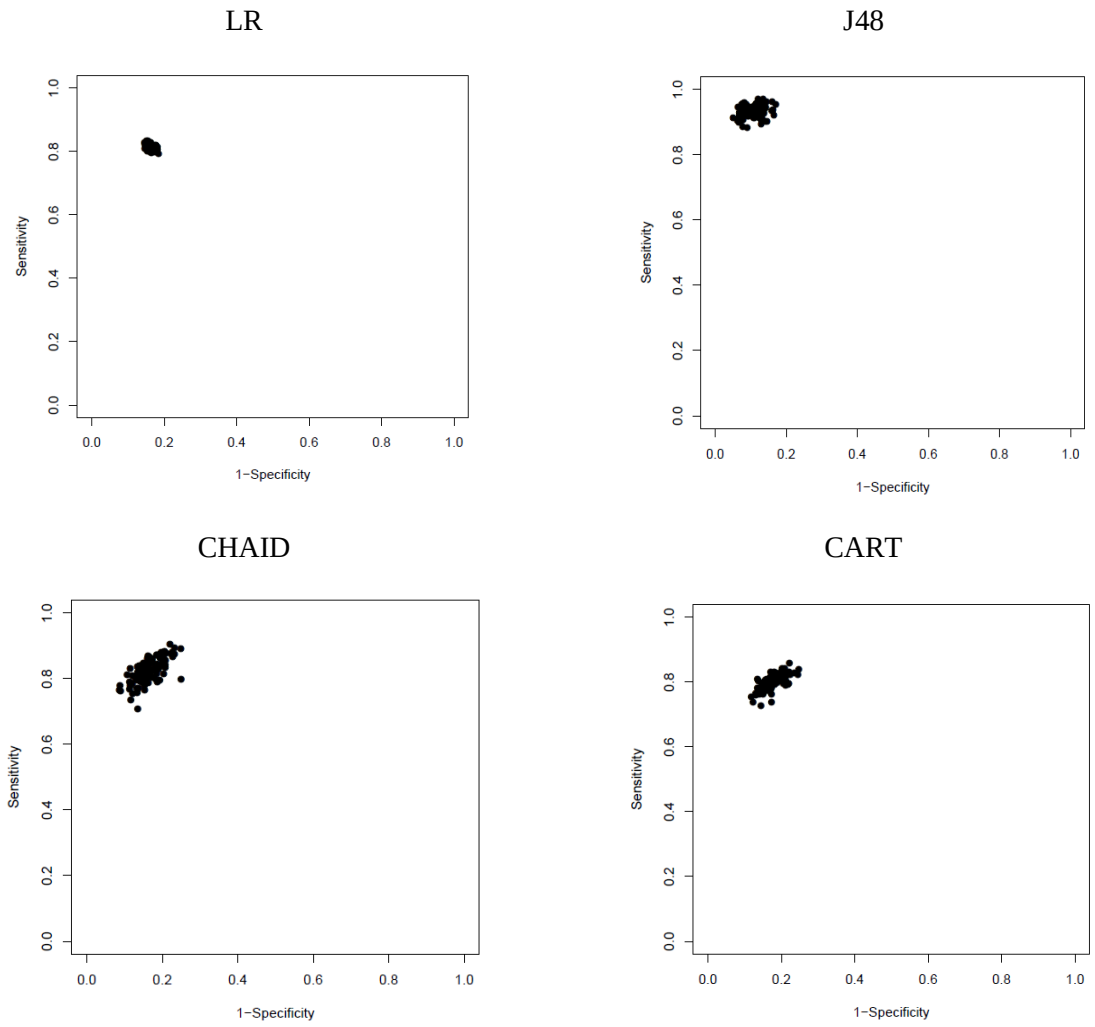
Şekil 26. Tümü sürekli yapıda bağımsız değişkenlerin 1000 denemelik simülasyon çalışması sonuçlarında AUC

Sürekli bağımsız değişkenlerin tümü kategorik 1000 denemelik çalışma sonuçlarında en yüksek AUC oranı N ve N-F dağılımlarında J48 yönteminde (0,96) gözlenirken en düşük AUC değeri F dağılımında CART yönteminde (0,79) gözlenmiştir (Şekil 26).

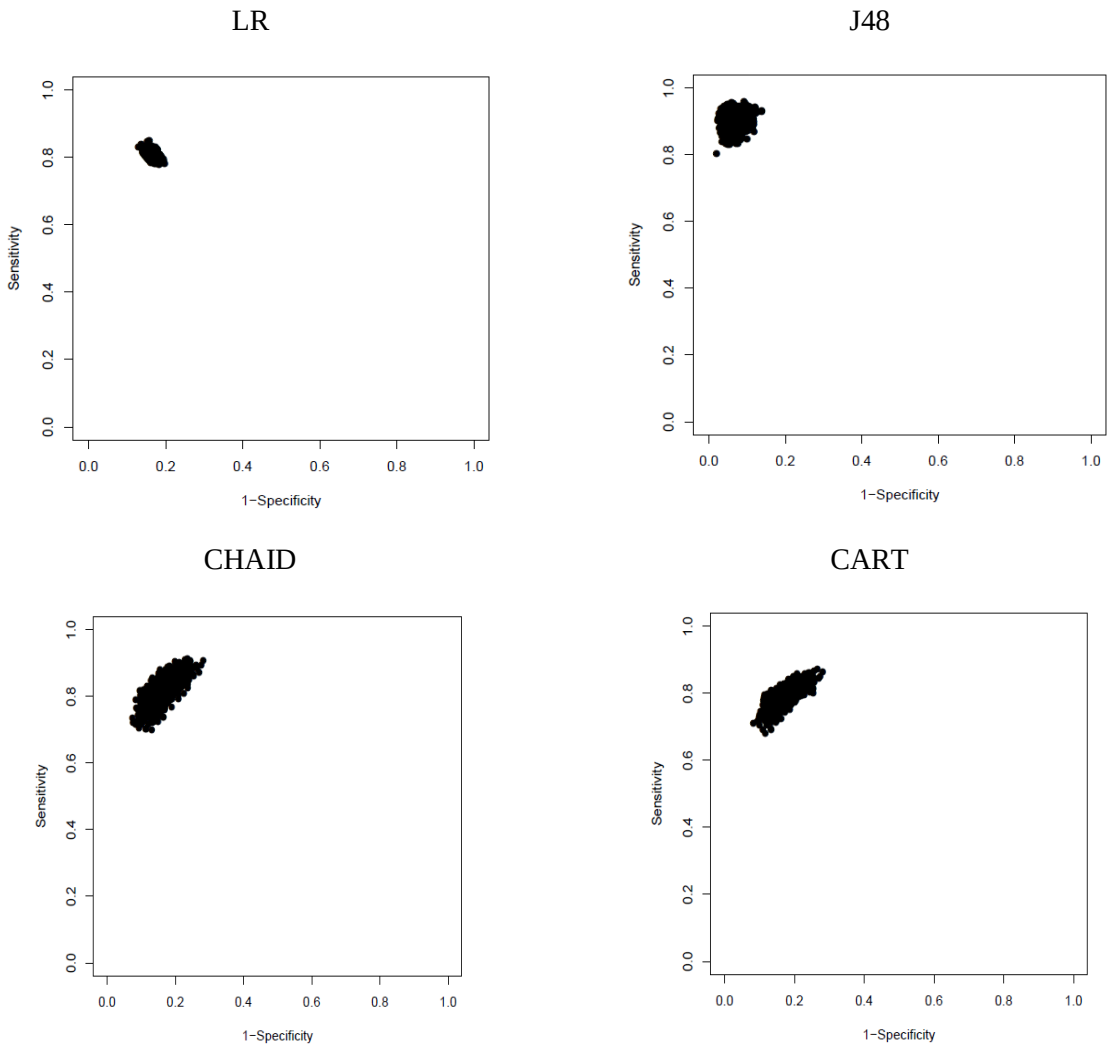
Tümü sürekli yapıda olan bağımsız değişkenler için yöntemlere göre Sensitivite değerlerine karşılık 1-Spesifite değerlerinin grafiksel gösterimi (Şekil 27-29)'da gösterildi (30, 100, 1000 deneme).



Şekil 27. Tümü sürekli yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-spesifite değerlerinin grafiksel gösterimi (30 deneme)



Şekil 28. Tümü sürekli yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-spesifite değerlerinin grafiksel gösterimi (100 deneme)



Şekil 29. Tümü sürekli yapıda olan bağımsız değişkenler için yöntemlere göre sensitivite değerlerine karşılık 1-spesifite değerlerinin grafiksel gösterimi (1000 deneme)

TARTIŞMA

Çalışmamızda karar ağaçlarından CART, CHAID ve C4.5 (Java uygulaması J48) ve LR analizinin performanslarını simülasyon verileri kullanarak karşılaştırıldı. Bağımsız değişkenin tümü kategorik olduğunda duyarlılık, özgüllük pozitif kestirim, negatif kestirim, doğruluk ve AUC kriterlerine göre doğru sınıflamada en düşük oran CART algoritmasında gözlenirken en yüksek oran J48 algoritmasında gözlenmiştir. Bağımsız değişkenin 5 kategorik 5 sürekli olduğunda duyarlılık, özgüllük, pozitif Kestirim, negatif kestirim, doğruluk ve AUC kriterlerine göre doğru sınıflamada en düşük oranı CART ve LR algoritmasında gözlenirken en yüksek oran J48 algoritmasında gözlenmiştir. Ve son olarak Bağımsız değişkenin tümü sürekli olduğunda duyarlılık, özgüllük, pozitif Kestirim, negatif kestirim, doğruluk ve AUC kriterlerine göre doğru sınıflamada en düşük oranı LR algoritmasında gözlenirken en yüksek oran J48 algoritmasında gözlenmiştir. C4.5 algoritmasının Java uygulaması olan J48 bağımsız değişkenin tüm durumlarında en doğru sınıfa atamada diğer algoritmalara göre üstün olduğu görülmüştür.

Son yıllarda karar ağacı yöntemlerinin sağlık alanında uygulamalarının yaygınlaştığı ve bu yöntemlerin performanslarının karşılaştırıldığı görülmektedir. Yapılan çalışmalarda karar ağaçlarının performansları duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC gibi kriterler kullanılarak karşılaştırılmıştır. Çalışmamızda karar ağacı yöntemlerinden olan

CART, CHAID ve C4.5 (java uygulaması J48) ile LR analizinin performansları simülasyon verileri kullanılarak karşılaştırılmıştır. Bu karşılaştırmalar doğrultusunda araştırmamızın temel bulguları söyle sıralanabilir: i) tümü kategorik yapıda olan bağımsız değişkenler için 30, 100 ve 1000 denemelik simülasyon çalışması sonuçlarına göre, dört yöntem arasında en düşük duyarlılık oranı CART algoritmasında gözlenirken diğer üç algoritmanın duyarlılık oranlarının birbirine yakın değer almışlardır, ii) 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 30, 100 ve 1000 denemelik simülasyon çalışması sonuçlarına göre, en düşük duyarlılık oranı LR yönteminde gözlenmiş olup en yüksek değer ise J48 yönteminde elde edilmiştir, iii) sürekli yapıda olan bağımsız değişkenler için 30, 100 ve 1000 denemelik simülasyon çalışması sonuçlarına göre, dört yöntem en düşük duyarlılık oranı LR yönteminde gözlenirken en yüksek değer ise J48 yönteminde gözlenmiştir.

Karar ağaçları yöntemleri konusunda literatürde sağlık alanında yapılan çalışmalar incelendiğinde Tang ve ark.'nın koroner kalp hastalığını etkileyen faktörleri (yüksek tansiyon, alkol kullanımı, cinsiyet, uyku ve ağrı gibi) belirlemede ID3, C4.5, CART ve CHAID karar ağacı yöntemlerini kullandığı ve performans karşılaştırmalarında en doğru sınıflama oranını C4.5 algoritmasının verdiğini bildirmişlerdir (50).

Süt ve arkadaşları kafa travmalarındaki ölümleri tahmin etmede kullandıkları karar ağaçları tekniklerinin karşılaştırılmasında CHAID algoritmasının CART algoritmasından doğruluk ve duyarlılık değerleri bakından daha üstün olduğunu bildirmiştir (51).

Türe ve arkadaşlarının çalışmalarında hipertansiyonu tahmin etmede sınıflandırma tekniklerinin performanslarını karşılaştırmışlardır. Çalışmada karar ağaçlarından (CART, QUEST, CHAID, MARS) ve istatistiksel sınıflandırma yöntemlerinden (FDA, LR analizi) kullanılarak söz konusu yöntemlerin sınıflandırma performansları karşılaştırılmıştır. Sonuç olarak CHAID analizi (CART, QUEST ve LR) diğer tekniklere göre hipertansiyon hastalığının tahminin etmede daha doğru sonuçlar verdiğini bildirmişlerdir (52)

Lemon ve arkadaşları halk sağlığı üzerinde yaptıkları çalışmada, riskli grupta olan ve benzer özellikleri gösteren hastaları sınıflandırmak için CART ile LR analizini kullanmış ve iki analizin sonuçlarını karşılattırırken CART analizinin sınıflandırmada daha umut verici sonuçlar ortaya koyduğunu bildirmiştir (53).

Türe ve arkadaşları 500 meme kanserli hasta üzerinde yinelemesiz sağ kalım süresini etkileyen risk faktörlerinin belirlenmesinde karar ağacı yöntemlerinden CART, CHAID, QUEST, C4.5 ve ID3 ile Kaplan-Meier analizlerini performans olarak birbiri ile karşılaştırılmış ve risk faktörlerini öngörmede C4.5'i diğer yöntemlere göre daha iyi olduğunu bildirmiştir (46).

Moraco ve arkadaşları Demans hastaları üzerinden yürüttükleri çalışmada veri madenciliği yöntemlerinden (Çok Katmanlı Algılayıcılar, Yapay Sinir Ağları, Radyal Taban Fonksiyonlu Sinir Ağları, Destek Vektör Makineleri, CART, CHAID QUEST ve Random Ormanlar) ayrıca geleneksel sınıflayıcılardan (Doğrusal Diskriminant Analizi, Kuadratik Diskriminant Analizini) ROC eğrisi altında kalan alan, doğruluk, özgüllük, duyarlılık açısından performanslarını karşılaştırmıştır. Çalışmada özgünlük bakımından CHAID algoritması CART algoritmasından daha üstün bir performansa sahip olduğu bildirilmiştir (54).

Delen ve arkadaşları meme kanserinde hayatta kalmayı tahmin etmede C4.5 algoritmasının doğruluk oranını (%93,6), Yapay Sinir Ağlarını (%91,2) ve LR doğruluk oranını (%89,2) olarak tespit etmişler bu doğruluk oranları göz önüne alındığında C4.5 in diğer tahmin edicilerden da üstün olduğu bildirilmiştir (55).

Coşkun ve arkadaşları veri madenciliği yöntemlerini doğruluk derecesi bakımından karşılaştırdıkları çalışmada sonuçlar incelendiğinde J48 algoritmasının model testine ait (%86,36) doğruluk derecesiyle en iyi sonucu ürettiğini LR ise (%85,6) doğruluk oranı ile J48 den sonra geldiği bildirilmiştir (56).

Goel ve arkadaşları Hintli ergenlerde insülin direncini tahmin etmede rutin klinik ve biyokimyasal parametrelere dayalı basit öngörü karar modellerle LR karşılaştırmıştır. 14-19 yaş aralığında aşamalı küme örneklemesi ile seçilmiş 793 ergenin insülin direncini doğru sınıflama probleminde CART algoritmasının duyarlılık, özgüllük, ROC eğrisi altında kalan alanları kriter edildiğinde daha başarılı olduğu tespit edilmiştir (57).

Trujillano ve arkadaşları yoğun bakımda kritik durumda olan hastaların ölüm istatistiklerinin sınıflandırma tahmininde CART, CHAID, C4.5 ve LR algoritmalarının sonuçlarını karşılaştırmıştır. Karşılaştırma CART, CHAID, C4.5 ve LR algoritmalarının doğruluk oranlarının karşılaştırılması şeklinde olmuştur. Sonuç olarak doğruluk oranları şu şekilde belirlenmiştir. C4.5 doğruluk oranı (0,78) ile en yüksek doğruluk oranına sahip olurken CHAID ve CART (0,75) tespit edilmiş son olarak LR doğruluk oranın (0,77) olarak bulunmuştur. Bu sonuçlara göre Trujillano ve arkadaşlarının yoğun bakım hastalarının ölüm tahminleri üzerine yaptıkları çalışmada C4.5 algoritmasının doğruluk oranının CART, CHAID ve LR algoritmalarından daha üstün olduğunu belirtmişlerdir (58).

Mani ve arkadaşları elektronik veri tabanından elde edilen bir meme kanseri verisini çeşitli karar ağacı algoritmalarını kullanarak hastalığın nüksetmesini tahmin etmek istemişlerdir. Bağımsız değişken olarak tümör varlığı, tümör büyüklüğü, tümör yayılma hızı gibi klinik veriler kullanılmış. Elde edilen sonuçlara göre tahmin araçlarının doğruluğu CART (%63,4), C4.5 (63,9), FOCL (%66,4) ve Bayes (%68,3) olarak bulunmuştur. Bu çalışmada bize doğruluk kriteri bakımından C4.5 in CART algoritmasından üstün olduğunu göstermektedir (59).

Mani ve arkadaşları 678 Alzheimer hastası üzerinde yaptığı değerlendirmede, hastaların hafıza, oryantasyon, karar verme ve problem çözme, kamu işleri, hobi ve kişisel bakım bağımsız değişkenlerini dikkate alarak yaptığı sınıflandırmada C4.5 ve CART algoritmalarını kullanmıştır ve bu algoritmaları doğruluk karşılaştırmasında C4.5 (%86,3),

CART (%82,9) bulunmuştur. Buna göre C4.5 algoritması doğruluk performansı bakımından CART algoritmasının önüne geçtiği belirtilmiştir (60)

Chen ve arkadaşları tanınmış sınıflandırma ağaçlarını (CART, Yapay Sinir Ağları, C4.5 Karar Ağacı, Bayes) kullanarak kanser genlerini tanımlamıştır. Yapılan çalışmada doğruluk bakımından karşılaştırıldığında C4.5 algoritması (%74) seviyelerinde CART algoritması ise (%70) seviyelerinde doğru sınıflama performansı gerçekleştirdiği bildirilmiştir (61).

Ploeg ve arkadaşları kafa travması geçirmiş 3181 hastanın 243 ünde tomografilerinde bulgu tespit edilmiş. Kafa travmasını sınıflandırmada LR, Bayes, CHAID, Yapay Sinir Ağları, CART algoritmalarını duyarlılık, özgünlük ve ROC eğrisi altında kalan alan kriterleri ile karşılaştırmıştır. Alınan sonuçlara göre Bayes ROC eğrisi altında kalan alan (0.806) LR algoritması ROC eğrisi altında kalan alanda (0.800), Yapay Sinir Ağları ROC eğrisi altında kalan alan (0.782), CHAID algoritmasında ROC eğrisi altında kalan alan (0.759), CART algoritmasında ROC eğrisi altında kalan alan (0.759) olduğu bildirilmiştir. Bu sonuçlara göre takip ettiğimiz algoritmalar birbirlerine çok yakın değerler gösterdiği gözlenmesiyle birlikte; LR algoritmasında ROC eğrisi altında kalan alan değerlendirildiğinde en iyi performansı göstermiş onu takip eden CHAID olmuştur, en düşük performans ise CART algoritmasında gözlendiği bildirilmiştir (62).

Yeon Ji ve arkadaşları travmatik beyin hasarına karar vermede bazı karar ağaçlarını kullanmış ve bunların performans değerlendirmelerini incelemişlerdir. Araştırmada karar ağaçlarından CART, C4.5 ayrıca LR ve Yapay Sinir Ağları kullanılmıştır. Sonuç olarak algoritmalar doğruluk bakımından karşılaştırıldığında performanslar CART (%72), LR (%74,6) ve C4.5 (%75,6) şeklinde olduğu bildirilmiştir (63).

SONUÇLAR

Çalışmamızda bağımsız değişkenler simülasyon yapılarak, Karar ağaçları yöntemlerinden CART, CHAID, J48 ve Regresyon modellerinden LR'un bağımlı değişkeni tahmin etmede performanslarını duyarlılık, özgüllük, PKD, NKD, doğruluk ve AUC kriterlerine göre karşılaştırıldı. Elde edilen sonuçlar şu şekilde bulundu:

- Tümü kategorik yapıda olan bağımsız değişkenler için 30, 100 ve 1000 denemelik simülasyon çalışması sonuçlarına göre, dört yöntem arasında duyarlılık Özgüllük Pozitif Kestirim, Negatif Kestirim doğruluk ve AUC en düşük oranı CART algoritmasında gözlenirken en yüksek oran J48 algoritmasında gözlenmiştir.
- 5 kategorik 5 sürekli yapıda olan bağımsız değişkenler için 30, 100 ve 1000 denemelik simülasyon çalışması sonuçlarına göre, dört yöntem arasında duyarlılık Özgüllük Pozitif Kestirim, Negatif Kestirim doğruluk ve AUC en düşük oranı CART ve LR algoritmalarında gözlenirken en yüksek oran J48 algoritmasında gözlenmiştir.
- Sürekli yapıda olan bağımsız değişkenler için 30, 100 ve 1000 denemelik simülasyon çalışması sonuçlarına göre, dört yöntem arasında duyarlılık Özgüllük Pozitif Kestirim, Negatif Kestirim doğruluk ve AUC en düşük oranı LR algoritmalarında gözlenirken en yüksek oran J48 algoritmasında gözlenmiştir.

ÖZET

Çalışmamızın amacı karar ağacı yöntemlerinden olan CART, CHAID ve C4.5 (Java uygulaması J48) ile Lojistik Regresyon analizinin performanslarını simülasyon verileri kullanarak karşılaştırılmasıdır. Simülasyon verileri oluşturulurken bağımsız değişkenler tümü kategorik, tümü sürekli ve hem sürekli hem kategorik şekilde oluşturulmuş ve her bir yapıdan 30'lu, 100' lük ve 1000'li denemeler şeklinde simülasyonlar yapılmıştır. Yapılan simülasyonlar R programı ile CART, CHAID, J48 ve Lojistik Regresyon yöntemleri ile analiz edilmiştir. Performans değerlendirmemizde duyarlılık, özgüllük, pozitif kestirim değeri, negatif kestirim değeri, doğruluk oranı ve ROC eğrisi altında kalan alan değeri esas alınmıştır. Yapılan simülasyon çalışmalarında; tümü kategorik yapıda olan bağımsız değişkenler için 1000 denemelik simülasyon çalışmasına göre, dört algoritma arasında en düşük duyarlılık oranı (%79.92) CART yönteminde gözlenirken diğer üç yöntemin duyarlılık oranlarının birbirine yakın değerler (J48-%85.89, CHAID-%85.00, Lojistik Regresyon-%82.50) aldığı bulunmuştur. 5 kategorik, 5 sürekli yapıda olan bağımsız değişkenler için 1000 denemelik simülasyon çalışması sonuçlarına göre, dört yöntem arasında sürekli değişkenlerden 3 değişkenin F dağılımından, 2 değişkenin normal dağılımdan türetilen bağımsız olan değişkenler göz önüne alındığında en düşük duyarlılık oranı Lojistik Regresyon yönteminde (%79,19) gözlenirken, CART yönteminde (%81,94), CHAID yönteminde (%84,85), en yüksek değer ise J48 yönteminde (%91,80) gözlenmiştir. Sürekli yapıda olan bağımsız değişkenler için 1000 denemelik simülasyon çalışması sonuçlarına göre, dört yöntem arasında sürekli değişkenlerden 3 değişkenin F dağılımından, 2 değişkenin normal dağılımdan türetilen bağımsız olan değişkenler göz önüne alındığında en düşük duyarlılık oranı Lojistik Regresyon yönteminde (%75,64) gözlenirken, CART yönteminde (%79,67), CHAID yönteminde (%84,75), en yüksek değer ise J48 yönteminde (%93,17) gözlenmiştir.

Sonuç olarak bağımsız değişkenin yapısı ve simülasyon deneme sayısı değişse de sonuçlarda dikkat çekici bir farkla J48 (C4.5 java uygulaması) yöntemi diğer yöntemlerden daha yüksek bir performans göstermiştir.

Anahtar Kelimler: CART, CHAID, C4.5 (J48), Lojistik Regresyon (LR), Simülasyon.

COMPARISON OF PERFORMANCES OF DECISION TREES AND LOGISTIC REGRESSION ANALYSIS BY A SIMULATION STUDY

SUMMARY

The aim of the study is to compare performances of CART, CHAID and C4.5 (java application J48) decision tree methods with Logistic Regression (LR) analysis by simulation data. In the simulation processes, independent variables were classified as all categorical, all continuous, both continuous and categorical, and they were simulated 30, 100 and 1000 trials. The simulations and analysis (CART, CHAID, J48 and LR methods) were done using the R program. Sensitivity, specificity, positive predictive value, negative predictive value, accuracy rate, and area under the ROC curve were used for performance evaluation. In accordance with simulations consisting of 1000 trials, while the lowest sensitivity rate among the four methods was observed in CART (79.92%), it was found that the sensitivity rates of the other three methods had closer rates to each other (J48-85.89%, CHAID-85.00%, Logistic Regression-82,50%) for all independent variables in categorical forms in simulation studies. According to the results of simulation of 1000 trials for 5 categorical and 5 continuous independent variables, it was observed that the lowest sensitivity ratio belonged to Logistic Regression (79,19%), CART method (81,94%), CHAID method (84,85%) and the highest ratio was in J48 (91,80%) when among four methods 3 variables of continuous variables derived from F distribution and 2 variables derived from normal distribution were taken into account. According to the results of simulation of 1000 trials for continuous independent variables, it was observed that the lowest sensitivity ratio belonged to Logistic Regression (75,64%), CART method (79,67%), CHAID method (84,75%) and the highest ratio was in J48 (93,17%) when among four methods 3 variables of continuous variables derived from F distribution and 2 variables derived from normal distribution were taken into account. As a result, though the structure of independent variable and the number of trials changed, J48 (C4.5 java application) turned out to perform considerably higher than the other methods in the results.

Key Words: CART, CHAID, C4.5 (J48), Logistic Regression (LR), Simulation.

KAYNAKLAR

1. Pehlivan G. Chaid Analizi Ve Bir Uygulama (tez). İstanbul: Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü; 2006.
2. Kıran ZB. Lojistik Regresyon Ve Cart Analizi Teknikleriyle Sosyal Güvenlik Kurumu İlaç Provizyon Sistemi Verileri Üzerinde Bir Uygulama (tez). Ankara: Gazi Üniversitesi Fen Bilimleri Enstitüsü; 2010.
3. Özkan Y. Veri madenciliği yöntemleri. Çölkesen R, Uğurkaya C (Editörler). İstanbul: Papatya Yayıncılık Eğitim; 2008. s.150–116.
4. Zaki MJ. Scalable Data Mining for Rules (tez). New York: University of Rochester Department of Computer Science; 1998.
5. Kocabaş FM. Veri Madenciliği Süreci Ve Gerçek Bir Veri Seti Üzerinde Uygulanması (tez). Ankara: Hacettepe Üniversitesi Fen Bilimleri Enstitüsü; 2010.
6. Ediz B. Lojistik Regresyon Ayırma Analizi, Ayırma Sorunu Ve Kalp Hastalarında Lojistik Model Yardımıyla Risk Ölçütlerinin Belirlenmesi (tez). Bursa: Uludağ Üniversitesi Sağlık Bilimleri Enstitüsü; 1997.
7. Silahtaroğlu G. Kavram ve algoritmalarıyla veri madenciliği. Çölkesen R, Uğurkaya C (Editörler). İstanbul: Papatya Yayıncılık Eğitim; 2008. s.87–80.
8. Gülpınar V. Avrupa Birliği Ülkeleri İle Türkiye'nin Ekonomik Göstergelerinin Karar Ağacı Yöntemi İle Karşılaştırılması (tez). İstanbul: Marmara Üniversitesi Sosyal Bilimler Enstitüsü; 2008.
9. Han J, Kamber M. Data mining concept and techniques. In: Cerra D (Eds.). University of illinois at urban. 2th ed. San Francisco: Morgan Kaufmann Publ; 2000.p. 468-460.

10. Ning P, Steinbach M. Introduction to data mining. In: Kumar V (Eds.). 1th ed. Boston: Addison-Wesley Longman Publ.2005.p.70-1.
11. Yıldırım P, Uludağ M, Görür A. Hastane bilgi sistemlerinde veri madenciliği. ÇOMÜ Akad Biliş Derg 2004;10:434-429.
12. Gölbaşı G. Sınıflama ve Regresyon Ağaçları ve Bir Uygulama (tez). İstanbul: Mimar Sinan Güzel Sanatlar Üniversitesi Fen Bilimleri Enstitüsü; 2000.
13. Berson A, Smith S, Thearling K. Building data mining for crm. In: Berson A (Eds.) New York: A Division of the McGraw-Hill Comp; 1999.p.50-1
14. Kandartzic M. Data mining consept models and algorithms. 2nd ed. Canada: John Wiley&Sons Co; 2011.p.140-130.
15. Quinlan JR. C4.5 programs for machine learning. In: Morgan MB, Sery D (Eds.). New York: Morgan Kaufman Publ;1993.p.66-57.
16. Dunham MH. Data mining in troductory and advanced topics. In: Hall P (Eds.). New Jersey: Pearson Education Co;2003.p.60-5.
17. Akgöbek Ö, Öztemel E. Endüktif öğrenme algoritmalarının kural üretme yöntemleri ve performanslarının karşılaştırılması. SAÜ Fen Bil Derg 2006;1:10-7
18. Yıldırım S. Tümevarım Öğrenme Tekniklerinden C4.5'in İncelenmesi (tez). İstanbul: İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü; 2003.
19. Quinlan JR. Induction of decision trees. Machine learning, 1th ed. Boston: Kluwer Academic Publ;1986.p.106-81.
20. Kass GV. An exploraty tecnique for investigating large quantaties of categorical data. Appl Statist 1980;29(2):127-119.
21. Smart Drill. Data mining, analytic techniques: chaıd, 2001. [Online] http://www.smartdrill.com/CHAID.html (Erişim Tarihi: 01.06.2013)
22. Özdamar EÖ. Veri Madenciliğinde Kullanılan Teknikler ve Bir Uygulama (tez). İstanbul: Mimar Sinan Güzel Sanatlar Üniversitesi Fen Bilimleri Enstitüsü; 2002.
23. Yohannes Y, Webb P. Classifications and regression trees, cart: a user manuel for identifying indicators of vulnerability to famine and chronic food insecurity. In: Bouis H, Haddad L (Eds.). Food , Washington; International food Inst Publ; 1999.p.34-1.
24. Quinlan JR. Simplifying decision trees. Int J Man-Machine Stud 1987;27(4):370-349.
25. Fiske J. Introduction to communication studies. 2th ed. New York: Routledge Publ; 1994;ch 37-35.

26. Colin A. Building decision trees with the ıd3 algorithm. Dr Dobbs J 1996;13:4-1.
27. Sastry PM, Krishnan R, Ram BVS. Classification and identification of teluguh and written characters extracted from palm leavesusing decision tree approach. ARPN J Eng Appl Scı 2010;5:3-1.
28. Koyuncugil AS, Özgülbaş N. Veri madenciliği: tıp ve sağlık hizmetlerinde kullanımı ve uygulamaları. Bilişim Tekn Derg 2009;2:21.
29. Prather JC, Lobach DF, Goodwin LK, Hales JW, Hage ML. Discovery ın a clinical data warehouse. Proc AMIA Annu Fall Symp 1997;105–101.
30. Akgöbek Ö, Kaya S. Veri madenciliği teknikleri ile veri kümelerinden bilgi keşfi: medikal veri madenciliği uygulaması. E J New World Sci Acad 2011;6:239.
31. Hu R. Medical data mining based on decision tree algorithm. Comput Inform Sci 2011;4:15-1.
32. Çolak C, Çolak MC, Orman MN. Koroner arter hastalığının tahmininde lojistik regresyon modeli seçim yöntemlerinin karşılaştırılması. Anadolu Kardiyol Derg 2007;7:11-6.
33. Albayrak A, Yılmaz KŞ. Veri madenciliği: karar ağacı algoritmaları ve imkb verileri üzerine bir uygulama. SDÜ İkt İdar Bil Fak Derg 2009;14:36-1.
34. Gökçen H, Özkil A, Yardımoğlu H. Peker D. Kamuda karar destek sistemlerinin kullanımı ve bir model önerisi. XII Türkiye Bilişim Dern Kamu Bilgi İşlem Merk Yön Birl Kamu Bilişim Platformu Özet Kitabı s.12, İstanbul,2010.
35. Demirel B. Veri Madenciliğinde Chaid Algoritmasının Sosyal Güvenlik Kurumu Veri Tabanına Uygulanması. (tez). Ankara: Gazi Üniversitesi Fen Bilimleri Enstitüsü; 2010.
36. Jeffrey WS. Data mining: an overview. CRS Report for Congress; 2004 Dec 16;Washington, America: Elsevier; 2004:4.
37. Vahaplar A, İnceoğlu MM. Veri madenciliği ve elektronik ticaret. www.bayar.edu.tr/bilisim/dokuman/inceoglu.doc (Erişim Tarihi: 05.12.2012).
38. Yaralıoğlu K. Veri madenciliği. [online]. 2009. http://www.deu.edu.tr/userweb/k.yaralioglu/dosyalar/ver_mad.doc (Erişim Tarihi: 27.12.2013).
39. Cheng JH, Chen HP, Lin YM. A hybrid forecast marketing timing model based on probabilistic neural network, rough set and c4.5. Expert Syst Appl 2010;37:18-15.
40. Fawcett T. An ıntroduction to roc analysis. Pattern Recogn Lett 2006:(10):861.

41. Tek Ö. Çocuk Suçluluğunun Chaid Çözümlemesi ile Değerlendirilmesi (tez). Ankara: Gazi Üniversitesi Sosyal Bilimler Enstitüsü;2012.
42. Yağız Z. CHAID analizi (tez). Ankara: Gazi Üniversitesi Fen Bilimleri Enstitüsü; 2003.
43. Kuzey C. Veri Madenciliğinde Destek Vektör Makinaları ve Karar Yöntemlerini Kullanarak Bilgi Çalışanlarının Kurum Performansı Üzerine Etkisinin Ölçülmesi ve Bir Uygulama (Tez). İstanbul: İstanbul Üniversitesi Sosyal Bilimler Enstitüsü İşletme Anabilim Dalı Sayısal Yöntemler Bilim Dalı; 2012.
44. Aydın S. Veri Madenciliği ve Anadolu Üniversitesi Uzaktan Eğitim Sisteminde Bir Uygulama (tez). Eskişehir: Anadolu Üniversitesi Sosyal Bilimler Enstitüsü; 2007.
45. Tüzüntürk S. Veri madenciliği ve istatistik. UÜ İkt İdr Bil Fak Derg 2010;29(1):90-65.
46. Türe M, Tokatlı F, Kurt I. Using kaplan–meier analysis together with decision tree methods (c&rt, chaıd, quest, c4.5 and ıd3) in determining ecurrence free survival of breast cancer patients. Expert Syst Appl 2009;36:2026-2017.
47. Çamdeviren HA, Yazici AC, Akkuş Z, Buğdaycı R, Sungur MA. Comparison of logistic regression model and classification tree: an application to postpartum depression data. Expert Syst Appl 2007;32:987.
48. Chudzinska M, Baralkiewicz D. Application of ıcp-ms method of determination of 15 elements ın honey with chemometric approach for the verification of their authenticity. Food Chem Toxıcol, 2011;49:2749-2741
49. Schubert CM, Thorsen SN, Oxley ME. The roc manifold for classification systems, Pattern Recogn 2011;44:362-350
50. Tang TI, Zheng G, Haung Y, Shu G, Wang P. A comparative study of medical data classification methods based on decision tree and system reconstruction analysis, IEMS J 2005;4:108-102.
51. Süt N, Osman Ş. Comparison of regression tree data mining methods for prediction of mortality ın head injury. Expert Syst Appl 2011;38(12):15–9.
52. Türe M, Kurt I, Kurum AT, Özdamar K. Comparing classification tecniques for predicting essential hypertension. Expert Syst Appl 2005;29:583-8.
53. Lemon SC, Roy J, Clark MA, Friedmann PD, Rakowski W. Classification and regression tree analysis ın public health: methodological review and comparison with logisitic regression. Ann Behav Med 2003;26(3):181-172.
54. Maroco J, Silva D, Rodrigues A, Savaşçı M, Santana I, Mendonça A. Data mining methods in the prediction of dementia: a real-data comparison of the accuracy,

sensitivity and specificity of linear discriminant analysis, logistic regression, neural networks, support vector machines, classification trees and random forests. BMC Res Notes 2011;4:299

55. Delen D, Walker G, Kadam A. Predicting breast cancer survivability: a comparison of three data mining methods. Artıf Intell Med 2005;34(2):127-113.
56. Coşkun C, Baykal A. Veri madenciliğinde sınıflandırma algoritmalarının bir örnek üzerinde karşılaştırılması. XIII. Akademik Bilişim Kongresi Bildirileri s.53, Malatya, 2011.
57. Goel R, Misra A, Kondal G, Pandey RM, Vikram DK, Wasir JS et al. Identification of ınsulin resistance in asian ındian adolescents: classification and regression tree (cart) and logistic regression based classification rules. Clın Endocrınol 2009;70(5):724-717.
58. Trujillano J, Badia M, Servia L, March J, Rodriguez A. Stratification of the severity of critically ıll patients with classification trees. BMC Med Res Methodol 2009;83(9):12-1.
59. Mani S, Pazzani JM, West MDJ. Knowledge discovery from a breast cancer database. Artıf Intell Med 1997;1211:133-130.
60. Mani S, Shankle WR, Dick MB, Pazzani JM. Two-stage machine learning model for guideline development. Artıf Intell Med 1999;16:71-51.
61. Chen KH, Wang ML, Tsai LM, Wanh KM, Adrian AM, Cheng WC, et al. Gene selection for cancer ıdentification: a decision tree model empowered by particle swarm optimization algorithm. BMC Bioinformatics 2014;15:2105-1471.
62. Ploeg VT, Smits M, Dippel DW, Hunink M, Steyerberg EW. Prediction of ıntracranial findings on ct-scans by alternative modelling techniques. BMC Med Res Methodol 2011;11:2288-1471.
63. Yeon JS, Smith R, Huynh T, Najarian K. A comparative analysis of multi-level computer-assisted decision making systems for traumatic ınjuries. BMC Med Inform Decis Mak 2009;9(2):1490-1472.

ŞEKİLLER LİSTESİ

TABLOLAR LİSTESİ

ÖZGEÇMİŞ

1981 yılında Erzincan'da doğdum. İlk, orta ve lise eğitimimi Erzincan'da tamamladım. 2002 yılında Ondokuz Mayıs Üniversitesi Fen Edebiyat Fakültesi İstatistik Bölümünde Lisans eğitimimi tamamladım. 2007 yılında Garanti Bankasında çalışmaya başladım ve halen aynı görevime devam etmekteyim. 2010 yılında Trakya Üniversitesi Sağlık Bilimleri Enstitüsü Biyoistatistik AnaBilim Dalında Yüksek Lisans eğitimine başladım.