İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

VERİ MADENCİLİĞİ TEKNİKLERİYLE KREDİ KARTLARINDA MÜŞTERİ KAYBETME ANALİZİ

YÜKSEK LİSANS TEZİ

Tuğba TOSUN

Tezin Enstitüye Verildiği Tarih : 4 Temmuz 2006
Tezin Savunulduğu Tarih : 8 Haziran 2006

Tez Danışmanı : Dr.Sabih ATADAN
Diğer Jüri Üyeleri Prof.Dr. Eşref ADALI
Doç.Dr. Coşkun Sönmez

MAYIS 2006

ÖNSÖZ

Bu çalışmada, bana yardım ve desteğini esirgemeyen danışmanım Dr.Sabih Atadan'a, konu seçimimdeki yardımları için Prof.Dr. Eşref Adalı'ya , tez için gerekli olan veriler ve üzerinde çalışacağım yöntemlerin belirlenmesinde bana yol gösterip değerli zamanını ayıran Sn.Mehmet Hamdi Özçelik'e, son olarak da tezim konusunda bana zorlu iki yıl boyunca manevi ve moral desteklerini sürekli gösteren aileme, Sn.Osman Feyzioğlu, Sn.Ufuk Marangozoğlu, Sn.Gülnur Eroğlu, Sn.Hilal Aldemir ve Sn.Onur Akatay'a teşekkür ediyorum.

Mayıs, 2006 **Tuğba TOSUN**

İÇİNDEKİLER

KISALTMALAR

Nd=X : Node = X, hedef negatif ve pozitif değerlerinin eşit olduğu yapraklardır.

TABLO LİSTESİ

ŞEKİL LİSTESİ

VERİ MADENCİLİĞİ TEKNİKLERİYLE KREDİ KARTLARINDA MÜŞTERİ KAYBETME ANALİZİ

ÖZET

Bu çalışmada Yapı Kredi Bankası'nın kredi kartı müşterilerinin çeşitli niteliklerdeki bilgileri incelenerek, kaybedilmiş bir müşterinin profili veri madenciliği yöntemleriyle ortaya çıkarılmaya çalışılacaktır.

Çalışmada, karar ağaçları yöntemi kullanılmıştır. İnceleme sırasında 30,000 adet müşterinin bilgileri üzerinde bu yöntemle çalışmalar yapılmıştır. Bu yöntemle ortaya çıkarılan kurallar test edilerek doğruluk oranları ortaya çıkarılmış, bunlar istatistiksel tablolarla gözönüne konmuştur.

Karar ağaçları genelde bu tip konularda sebep ortaya çıkartma konusunda oldukça güçlüdürler. Bu sebeple, çalışmanın en son bölümünde kural tabloları incelenerek, müşteri kayıplarının sebepleri ve ne zaman gerçekleştiği bilgisine ulaşılmaya çalışılacaktır.

Algoritmanın uygulanması için hızlı ve güvenilir olması sebebiyle C programlama dili seçilmiştir. Aynı zamanda bu algoritma, sonradan değiştirilmeye ve yeni eklemeler yapılmasına izin verecek şekilde tasarlanmaya çalışılmıştır.

CREDIT CARD CUSTOMER CHURN ANALYSIS WITH DATA MINING TECHNIQUES

ABSTRACT

In this study, some data attributes of credit card customers of Yapı Kredi Bank are investigated and the churned customer profile is stated by using data mining methods.

The decision tree method is used in the study. Total number of customer data, which is examined through this analysis, is 30.000.The rules obtained from the analysis are tested on the test data and the error and correctness rates are found and statistically measured.

Decision tree algorithms are powerful to find the causes of certain problems associated with their human-readable rule tables. These rule tables are examined to learn the cause, and time of customer churn.

The C programming language is chosen to apply the algorithm, as this language is relatively faster and reliable. The source code is designed to be modular, to be improved for further analysis

1. GİRİŞ VE ÇALIŞMANIN AMACI

Günümüzde bir çok firma için müşterinin kaybedilmesi, sadakatinin ölçülmesi ve geri kazanımın yollarının aranması oldukça popüler konular haline gelmeye başlamıştır. Firmalar, yeni müşteriler elde etmek yerine, halihazırdaki müşterilerini kaybetmemek için çeşitli çalışmalar ve kampanyalar düzenlemektedirler.

Türkiye'de bu konu, özellikle müşterinin kaybedilmesinin kolay gerçekleştiği bankacılık sektöründe çeşitli araştırmalara önayak olmuştur. Bankacılıkta çok önemli olan kredi kartı müşterilerinin kaybedilmemesi için kaybedilen müşterinin profilinin belirlenip, henüz kaybedilmemiş olanlar üzerinde bu analizden elde edilen bilgi neticesinde çalışmalar yapılmaktadır.

Burada yapılan çalışmanın amacı, kredi kartı müşterilerinin kaybedilme sebeplerinin bulunması için veri madenciliği yöntemlerinden faydalanarak sonuçlara ulaşmaktır. Böylece, müşterinin neden kaybedildiği bilgisinin yanı sıra, hangi tür müşterilerin daha sık kaybedildikleri tahmin edilmeye çalışılacaktır.

2. KREDİ KARTLARINDA MÜŞTERİ KAYBETME

2.1. Kredi Kartları Tanımı

Kredi kartı, kart sahibine harcamalarında nakit para ödemeksizin mal veya hizmet satın almalarını veya nakit para çekebilmelerini sağlayan bir ödeme aracıdır. Dünya genelinde kredi kartları 1960'lı yıllarda kullanılmaya başlanmış, son 20 yılda giderek önem kazanarak yaygınlaşmıştır.

Türkiye'de kredi kartlarının yaygın olarak kullanımı 1990'lı yıllarda başlamıştır. Özellikle son birkaç yıldır, Türkiye'de kredi kartı kullanımının hızlı bir şekilde yaygınlaştığı gözlenmektedir. Bankalararası Kart Merkezi A.Ş.'nin (BKM'nin) verilerine göre, Türkiye'de 1997 yılı sonunda toplam 4.847.166 kredi kartı bulunurken, bu sayı 2002 yılı sonunda üç kattan fazla bir artış ile toplam 15.743.064'e ulaşmıştır [5].

Kredi kartı, kart sahibi için kart sahibi açısından kısa vadeli bir kredi kaynağıdır. Kişiye, borcunu, bütçesine ve kendi planına göre taksitler halinde ödeme olanağı sağlar. Bir çok ülkede geçerli olduğundan büyük seyahatlerde büyük kolaylıklar sağlar. Ayrıca kredi kartı sahibi, kartı çıkaran kuruluşun sunduğu bazı özel nitelikteki hizmetlerden yararlanabilir.

Kredi kartları, bankalar için de oldukça karlı ürünlerdir. Bu kar payının çoğunu yüksek orandaki faizler oluşturur. Bankalar için kredi kartlarından kazanılan kar, tüm ürünlerin kar payı toplamının tek başına ¾'ü kadardır. Ayrıca, bazı bankalar yıllık kredi kartı aidatları alırlar. Geç ödemeden kaynaklanan faizler, ATM aidatları, limit aşımı cezaları ve POS makinasının bulunduğu işyerlerinden her çekimde alınan ücretler kredi kartlarının bankalar için kar olanaklarıdır.

2.2. Müşteri Kaybetme Tanımı (Customer Churn)

Eğer bir müşteri, ilgili firmayla üyelik anlaşmasını sonlandırırsa ve başka bir rakip firmanın müşterisi haline gelirse bu müşteri kaybedilmiş müşteridir [3].

Müşteri kaybetme, müşterinin sadakati ile oldukça yakından ilgilidir. Şu sıra, müşteri sadakatini sağlamanın tek yolunun fiyat düşürmekten geçmediği bir ekonomik dönem yaşanmaktadır. Buna göre, ürüne yeni değerler ekleyerek sadakati sağlama, bir çok endüstride bir norm haline gelmeye başlamıştır [6].

Müşteri kaybedilmesinde odaklanan nokta, bir firma için hangi müşterinin kaybedilme olasılığının yüksek olduğunun bulunması ve bu müşterilerin geri döndürülmesi için yapılacak çalışmanın maliyetinin analiz edilmesidir. Analizi yaparken en önemli nokta ise, kayıp olan müşterinin tanımının yapılmasıdır. Bazı durumlarda, bu tanımın yapılması oldukça zordur. Örneğin, bir kredi kartı müşterisi, kolaylıkla başka bir bankanın kredi kartını kullanmaya başlayabilir, ve bağlı olduğu bankanın kartını kapattırmaz. Bu durumda bu müşterinin kaybedildiğini anlamak için örneğin kullanım oranındaki düşüşe bakılabilir [6].

Müşterinin kaybedilmesi, kaybın çok kolay gerçekleştiği firmalar için önemli bir sorundur. Örnek olarak bankalar, sigorta şirketleri ve telekomünikasyon firmaları verilebilir [2].

Firmalar için yeni müşteriler kazanmanın maliyeti günden güne artmaktadır. Bu sebeple, pazarlama sektöründe yeni bir devir başlamıştır. Buna göre, firmalar, yeni müşterileri kazanmak için kampanyalar düzenlemek yerine, sahip oldukları müşterileri kaybetmemek için çeşitli programların arayışı içindedirler. Bunun tek yöntemi ise, müşteriyi kaybetmeden önce önlem almaya başlamaktır.

Tam bu noktada, müşteri kaybının modellenmesi, önemli bir rekabet şansı ve yeni bir çalışma alanı doğurmuştur. İyi bir modelleme ile, firma hangi müşterinin kaybedilmeye yakın olduğunu, hagisininse sadık olduğunu ortaya çıkartır. Modellemenin bir kısmı, müşteri değerinin ölçümlenip belirlenmesinden oluşur [6].

2.3. Kredi Kartlarında Müşteri Kaybetme

Kredi kartı hizmetleri veren bankalar için kar getiren müşteri, ödemelerini belirli dengelerde yapan müşteridir. Müşteriler, hesaplarını kapatmadan harcamalarını düşürürlerse, o zaman “sessiz kaybedilme” gerçekleşir. Her ay düzenli ödeme yapan müşteriler, banka için ancak belirli miktarların üzerinde harcamalar yaptıkları sürece kar getirirler.

Müşteri kaybının önlenebilmesi için yapılan modellerde veri madenciliği teknikleri kullanılır. Böylece, bir müşterinin kaybedilip edilmeyeceği ve ne zaman kaybedilebileceği ölçümlenir ve müşterilerin neden kaybedildiği ortaya çıkartılır.

2.4. Veri Madenciliği Tanımı

Özellikle büyük firmaların veritabanlarında bulundurdukları ham verilerin ölçüleri çok büyük bir hızla artmaktadır. Ancak ham veri, tek başına çok büyük bir bilgi içermez. Bugünün rekabete açık iş dünyasında, şirketlerin bu verileri, çok kısa zamanlarda, pazarlama, yatırım ve yönetim stratejilerine rehberlik etmek üzere belirli görüşlere ve işe yarar bilgiye dönüştürmesi gerekmektedir.

Veri madenciliği, bahsedilen bu türdeki çok büyük veri tabanlarındaki ya da veri ambarlarındaki veriler arasında bulunan ilişkiler, örüntüler, değişiklikler, sapma ve eğilimler, belirli yapılar gibi ilginç bilgilerin ortaya çıkarılması ve keşfi işlemidir.

Veri madenciliğinde kullanılan yöntem ve araçlar, çok kısa zamanlarda işin niteliğine yönelik stratejik soruları cevaplamada yardımcı olurlar. Ham veride gizli kalmış olan örüntüleri ve ilişkileri tahmini bilgilere dönüştürebilirler [2].

Veri madenciliği, henüz geliştirilmekte olan bir konu olsa da, geçmişe yönelik bilgiden avantaj elde etmek isteyen perakende satış, finans, sağlık, nakliyat, havacılık firmaları ve özellikle de bankalar, veri madenciliği araçlarını kullanmaktadırlar. Örüntü tanımlama teknolojileri, matematiksel ve istatistiksel teknikler ile, veri madenciliği bu tür kurumlara farkedilmeyen ilişkileri, eğilimleri, beklentileri veya anomalileri ortaya çıkarmada yardımcı olur.

Veri madenciliği, iş dünyasında aşağıdaki konularda sıklıkla kullanılır :

1. *Pazar payı kesimleme :* Bir firmadan, aynı ürün/hizmetleri satın alan müşterilerin ortak özelliklerini ortaya çıkartır.
2. *Müşteri Kaybetme :* Hangi müşterilerin firmadan ayrılıp başka bir rakip firmaya geçebileceklerini tahminler.
3. *Dolandırıcılığı Ortaya Çıkarma :* Hangi müşterisel hareketlerin dolandırıcılıkla ilgisi olabileceğini ortaya çıkartır.
4. *Direkt pazarlama :* Müşteri kazanmak için yapılan kampanya listelerinde hangi tür müşterilerin geri dönüşlerinin fazla olabileceğini ortaya çıkartır.

5. *İnteraktif pazarlama :* Bir web sayfasında gezen kişinin en fazla ilgili olabileceği alanları ortaya çıkartır.
6. *Pazar sepeti analizi :* Hangi ürünlerin büyük oranda beraber satın alınma olasılığının yüksek olduğunu ortaya çıkatır.
7. *Eğilim Analizi :* Tipik bir müşterinin belirli zamanlarda davranış farklılıklarını inceler.

Veri madenciliği teknolojisi, aşağıdakileri kullanarak yeni iş alanları ortaya çıkartabilir :

1. *Eğilim ve davranışların tahminlenmesi :* Veri madenciliği, büyük bir veritabanındaki tahminlenebilir bilginin ortaya çıkarılmasında yardımcı olur. Örneğin, bir iflası tahminleme, ya da bir popülasyonda hangi kesimin bazı olaylara karşı ortak davranış şekillerini göstereceğini tahminleme gibi.
2. *Önceden bilinmeyen örüntülerin keşfedilmesi:* Veri madenciliği araçları, büyük veritabanlarında gizli kalmış örüntüleri ortaya çıkartırlar. Örneğin, birbiriyle alakasız gibi gözüken iki ürünün aynı anda alınma olasılığının keşfi, veya kredi kartlarındaki hareketlerden dolandırıcılığın teşhis edilmesi gibi.

Veri madenciliğinin keşif ve tahminleme yapma tekniğine modelleme adı verilir. Modelleme, bilinen cevapların olduğu durumlardan kurallar ve sonuçlar çıkarılarak, cevapları bilinmeyen ortamlarda bu kural ve sonuçların veri üzerinde uygulanmasıdır.

Veri madenciliğinde tahminleme ve tanımlama için aşağıdaki adımlar uygulanır [1] :

1. Veriyi temizleme
2. Veri dönüştürme
3. Veri küçültme
4. Veri örnekleme
5. Veri madenciliği
6. Bilgi sunumu

2.4.1. Veri Temizleme

Pratikte, ortamdaki veri eksik, gürültülü, yanlış veya devamlılık göstermiyor olabilir. Veri temizleme rutinleri, eksik değerleri tamamlamak, gürültüyü veya hatalı verileri düzeltmek için kulllanılır [1] :

2.4.1.1. Eksik Değerler

Eksik değerler üzerinde uygulanacak metodlar şunlardır [1] :

a. Eksik değerleri analizden çıkartmak : Bu yöntem, eğer çok fazla nitelikte eksik değerler varsa, verimlilik sağlamaz.

b. Eksik değerleri elle doldurmak : Bu yöntem genelde zaman alıcıdır ve büyük veritabanları için yapılabilir değildir.

c. Eksik değerler için global bir değer belirlemek : Eksik nitelikler için global değişken vermek, analiz programının bu değeri sıklıkla kullanıldığı için bir değer olarak kabul edip hesaplamalara dahil etmesini sağlayabileceğinden yanlış analiz sonuçları ortaya çıkartabilir.

d. Eksik değerler içeren niteliğin ortalama değeri ile veriyi tamamlamak : Genelde, nitelik bir gelir bilgisiyse , ortalama kullanmak doğru sonuçlar verebilir.

e. En olası değer ile eksik veriyi tamamlamak : Bu metod için regresyon, Bayesian-formalism ya da karar ağaçları kullanılabilir.

2.4.1.2. Gürültülü Değerler

Gürültü, bir değişkendeki rastlantısal hata oranıdır. Verideki gürültüyü yoketmek için aşağıdaki yöntemler kullanılır [1] :

a. Kutulama : Bu yöntemde, veriler, komşu değerlerine göre sıralanırlar. Sıralanmış bu veriler, belirli sayıda “kutulara” konur. “Ortalama değere göre kutulama”da her değer, o kutuya dahil edilen tüm değerlerin ortalama değeri ile yer değiştirir. Aynı şekilde “Medyan değerine göre kutulama”da değerler medyanla, “Limitlere göre kutulama”da ise, değerler maksimum ya da minimum değere yakınlıklarına göre bu iki değerle yer değiştirirler.

b. Demetleme : Birbirine benzer değerler gruplara yada demetlere bölünerek her bir demetin sınır çizgileri belirlenir.

c. Regresyon : Veri, bir fonksiyona sokularak, o fonksiyon üzerine yerleşmesi sağlanır ve böylece gürültülü değerler otomatik olarak elenmiş olurlar.

2.4.2. Veri Dönüştürme

Veri dönüştürmede amaç, veriyi analiz edebilmek için uygun hale getirmektir. Bunun için aşağıdaki yöntemler kullanılabilir [1] :

a. Düzleştirme : Verideki gürültüyü azaltmak için kullanılır. Kutulama, demetleme ve regresyon bu gruba girer.

b. Birleştirme : Veriyi özetleme ve toparlama için kullanılan yöntemdir. Örneğin günlük satış verileri, aylık veya yıllık olarak tutulabilir.

c. Genelleştirme : Düşük seviyeli veriler, daha yüksek seviyelerdekilere dönüştürülerek hiyerarşik hale getirilebilir. Örneğin, sokak adı bilgisi yerine şehir, hatta ülke adı kullanılması, ya da rakamsal yaş değerleri yerine "genç", "orta yaşlı" veya "yaşlı" değerlerinin kullanılması gibi.

d. Normalizasyon : Değişken veriyi aralıklar şeklinde tutmak için kullanılır.

e. Yeni nitelik oluşturma : Veriyi daha iyi analiz edebilmek için, ona yeni nitelikler ve değerler eklenmesi işlemidir.

2.4.3 Veri Küçültme

Veriyi analiz ederken bazen çok büyük veriler analizi olumsuz yönde etkileyebilir. Bu sebeple verinin boyutunun küçültülmesi denenebilir. Bunun için en fazla kullanılan yöntemlerden biri histogramlardır. Histogramlar, veri dağılımlarını yaparken kutulama yöntemini kullanırlar ve veri miktarını verimli olabilecek hale getirerek azaltırlar. Veriyi bölümlerken histogramlarda çeşitli kurallar kullanılabilir [1] :

a. Eşit genişlikli : Verinin konduğu her kutunun genişliği veya aralık değeri aynıdır.

b. Eşit derinlikli (Eşit boylu) : Her kutudaki frekans dağılımı yaklaşık aynıdır. Yani eşit sayıda veri içerir.

Veri küçültmede kullanılan başka bir yöntem demetlemedir. Demetleme, veriyi gruplara ayırır. Bir grubun içindeki değerler birbirine yakınken, diğer gruplardakilere

o kadar uzak olmalıdırlar. Bir demetin kalitesi, onun çapı ile ölçümlenebilir. Bu çap, bir demetin içindeki iki objenin birbirine olan maksimum uzaklığına eşittir.

Veriyi küçültürken kullanılan en son yöntem örneklemedir. Örnekleme, büyük miktardaki verileri çok daha küçük boyutlarda sunmaya olanak sağlar. Çeşitleri şunlardır :

a. Yerine konmasız rastgele örnekleme : N adet değerden n tanesinin rastgele seçilmesi ile oluşturulur, bu durumda her değerin seçilme olasılığı 1/N'dir ve birbirine eşittir.

b. Yerine konmalı rastgele örnekleme : Bu örnekleme çeşidi, yerine konmasız örneklemeye benzer ancak bu kez, seçilen her değer, yeniden seçilebilmek üzere genel verinin içine yeniden konur.

2.4.4. Veri Madenciliği

Bir veri madenciliği operasyonunda farklı veri madenciliği teknikleri kullanılabilir. Her tekniğin kendine göre avantaj ve dezavantajları vardır. Bunların en fazla kullanılanları, karar ağaçları, sinir ağları, demetleme algoritmaları, k-nearest algoritmaları, genetik algoritmalar, bulanık mantık, ve bağ analizi olarak sayılabilir [1].

2.4.4.1. Sinir Ağları

Sinir ağları, beynin çalışmasını taklit ederek, analizde öğrenme gerçekleştikten sonra diğer gözlemlerden ortaya çıkarılacak sonuçlarla modelleme yapabilme yöntemleridir. Giriş değerlerinden çeşitli kuralları öğrenirler ve örüntüleri ortaya çıkartarak parametreleri yeni veri üzerinde uygularlar. Sinir ağları, tahminlemelerde, kredi puanlamada ve risk analizinde oldukça faydalıdır.

Bir sinir ağının yapısında girişi, çıkışı ve işlem bölümleri bulunan düğümler bulunur. Bir değeri tahminlemede kullanılmak üzere bir model oluşturabilmek için bu düğümlerin giriş değerleri çeşitli şekillerde kombine edilir. Her bir düğümün değeri, onu besleyen diğer düğümlerin toplam ağırlıklarından yola çıkılarak hesaplanır. Bir modeli oluşturmada önemli olan, doğru sonuçları ortaya koyabilecek uygun bağlantı ağırlıklarını bulabilmektir.

Sinir ağları, bir verideki ilişkileri ve örüntüleri ortaya çıkartmak için kullanılırlar. Bu veri, bir Pazar araştırmasının sonuçları ya da farklı koşullarda bulunan bir üretim işleminin ortaya çıkan sonuçları olabilir. Kullanıldığı alan ne olursa olsun, sinir ağları geleneksel yöntemlerin dışında işlemler yaparlar. Geleneksel yöntemde, analizi yapan kişi, bilgisayara durumu ve kuralları tek tek tanıtarak öğretir. Sinir ağlarının bu kodlamaya ihtiyaçları yoktur. Sadece ham veriyle, ve iyi bir test sürecinden sonra, ortaya tahminleme yapabilecek durumda olan bir analiz programı çıkartırlar [1].

2.4.4.2. Karar Ağaçları

Karar ağacı, veriyi sınıflandırma ve tahminleme yapmada kullanılan popüler bir veri madenciliği tekniğidir. Karar ağaçlarının sinir ağlarından daha çekici olan tarafı, onlardan farklı olarak ortaya kurallar çıkartabilmeleridir. Bu kurallar, kullanıcıların kolayca anlayabileceği şekilde ifade edilirler, bu da analizi kolay bir hale getirir.

Karar ağaçları, ağaç şeklinde sınıflandırıcı bir yapıdır ve ağaçtaki her düğüm ya bir yaprak düğümü ya da karar düğümünü simgeler. Karar ağaçları ile ilgili daha detaylı bilgiler ilerki bölümlerde verilecektir [1].

2.4.4.3. K-Nearest neighbour algortimaları

Bu algoritmalar, aslında demetlemenin bir çeşididir. K değeri, komşu olan kayıtların sayısını simgeler. Verilen N adet prototip örüntüye ve bunların doğru sınıflandırılmasına göre, algoritma sınıflandırılmamış olan bir örüntüyü en yakın komşu gruba bağlar. Sınıflandırmanın doğruluğu, k değerinin artmasıyla artış gösterir. Ancak bu yöntem, eğer veri gürültülüyse hatalı sonuçlar ortaya koyar. Ayrıca, geçmişe yönelik veriye ihtiyaç duyulur [1].

2.4.4.4. Genetik algoritmalar

Genetik algoritmalar, evrimsel gelişimi taklit ederek çalışırlar. Optimum çözüme mutasyon ve seçme yöntemiyle ulaşılır. Yüksek uygunluğu olan çözümler seçilir ve mutasyona uğratılarak daha yüksek uygunluklu çözümler üretmek üzere yeniden kullanılır. Genelde genetik algoritmalar, iş tarifelerinde veya motor dizaynlarında kullanılırlar [1].

2.4.4.5. Bulanık mantık (Fuzzy Logic)

Kesin değişkenler yerine olası değişkenlerin kullanıldığı yöntemdir. 0 ile 1 arasında değişen değer, o niteliğin kesinliğe ne kadar yakın veya uzak olduğunu belirler. Genelde kontrol sistemlerinde kullanılırlar [1].

2.4.4.6. Bağ analizi (Link Analysis)

Bir verinin içindeki bağları ortaya çıkartan yöntemdir. Genelde ürün ve müşterinin arasındaki bağların ortaya çıkartıldığı market sepeti analizinde, hedefe yönelik pazarlama alanında veya stok fiyatlarının değişimlerinde kullanılır [1].

2.4.4.7. Diğer Teknikler (OLAP)

OLAP, bilgiyi çoklu boyutta ve hiyerarşide sunabilmek için kullanılır. Örneğin çok büyük boyutlardaki satış istatistik verileri arasında, kullanıcılara hangi ürünlerin satış oranlarının daha fazla değişkenlik gösterdiğini sunabilir. Kullanıcıların, daha önemli olan değişkenlere ve değerlere odaklanmasını sağlar.

OLAP ve veri madenciliği arasındaki en büyük fark, OLAP'ın kullanıcı odaklı olması yani analiz yapan kişinin bir hipotez öne sürerek OLAP programını kullanıp bunu test etmesi; ancak veri madenciliğinde bunun tam tersi olarak, veri madenciliği progamının kendisinin bir hipotez oluşturmasıdır. Böylece bir veri madenciliği aracı, kullanıcının büyük bir veri kaynağında görmesinin çok zor olabileceği bir örüntüyü ortaya çıkartabilir. OLAP ise daha çok bütünleştirmeler, hesaplamalar ve sonuç karşılaştırmalarını grafiksel ortamda incelemeye olanak tanır [1].

2.5. Veri Madenciliğinin Zorlukları

Veri madenciliği, veritabanları, yapay zeka ve istatistik gibi farklı disiplinleri entegre etme gereksinimi doğurur. Bu sebeple, genelde yüksek performanslı bilgisayarlar ve uzman kullanıcılar tarafından analiz süreci gerçekleştirilir.

Her bir işlem için doğru olan tekniğin kullanılması şarttır, aksi takdirde eğer veri temizlenmezse, doğru toplanmazsa ve iyi analiz edilmezse, ortaya beklenmeyen veya hatalı sonuçlar çıkabilir.

Veri kaynağının genelde çok büyük olması da sistemsel, donanımsal ve zamansal açıdan çeşitli sorunlar doğurabilir. Veri tabanlarının büyüklükleri giderek artan bir

yapıda olduğundan, sistemlerin bu büyümeyi kaldırabilecek şekilde tasarlanmış olması gerekmektedir.

2.6. Çalışmanın Kapsamı

Bu tezde, bir bankaya ait olan kredi kartı müşterilerinin verileri incelenerek, kaybedilme oranları ve sebepleri tahminlenmeye çalışılacaktır.Bunun için, veri madenciliği yöntemleri kullanılacaktır.

3. KURAMSAL ÇALIŞMA

3.1. Karar Ağacı Algoritması Tanımı

Karar ağaçları, ağaç şeklinde sınıflandırıcılardır. Bu ağaçtaki her düğüm bir yaprağı veya karar düğümünü belirtir. Yaprak düğümü hedef niteliğin değeridir. Karar düğümü ise, bir nitelikte uygulanacak olan test değeridir, bu düğümü, o niteliğe ait olan tüm olası nitelik değerleri izler, bu değerler ise ağacın dallarını oluşturur.

Karar ağacı, bir örneği, kökten yaprağa kadar inceleyerek sınıflandırır. Karar ağaçlarının öğrenme algoritmaları, bir hipotezi sunmak için bir küme karar ağacı kullanırlar. Öğrenme kümesinde, ham veri incelenerek mümkün olan en iyi şekilde sınıflandırılır. Algoritma bu işlemi recursive olarak tekrar eder ve en son ortaya çıkardığı karar ağacı en son hipotezi oluşturur. İdeal olan karar ağacı, öğrenme kümesi dışındaki verilerde de aynı kuralları oluşturur ya da az hata payıyla aynı hipotez sonuçlarını ortaya çıkartır [4].

Aşağıda, bir karar ağacı algoritması gösterilmektedir [1] :

Algortima: Karar-Ağacı-Oluştur

Giriş Değişkenleri : *Örnek kümesi, nitelik-listesi*

Metod :

1. N düğümünü oluştur
2. Eğer *örnek* değerlerinin tümü aynı sınıftaysa (Sınıf C)
    i. N düğümünü C etiketinde bir yaprak düğümü olarak döndür
3. Eğer *nitelik-listesi* boşsa
    i. N düğümünü *Örnek* kümesindeki en fazla kullanılan sınıf etiketinde bir yaprak düğümü olarak döndür
4. *Nitelik-listesi'*nden en fazla bilgi kazançlı *test-niteliği'*ni seç
5. N düğümünün etiketine *test-niteliği* ismini ver
6. *Test-niteliği*'nin her bir bilinen değeri için dön
    i. N düğümünden *test-niteliği*=a_i olacak şekilde bir dal çiz
7. s_i = *örnek* kümesinde *test-niteliği*= a_i olan örnekler olsun
8. Eğer s_i boşsa

i. *Örnek* kümesi içinde en fazla kullanılan sınıf etiketinde bir yaprak çiz

9. Değilse *Karar-Ağacı-Oluştur(s_i, nitelik-listesi, test-nitelik)'* dan dönen düğüm değerini ağaca ekle.

Yukarıda belirtilen algoritma, ID3 algoritmasının bir versiyonudur, ve karar ağacı indirgenmesi (decision tree induction) olarak da bilinir. Temel strateji şu şekildedir [1].

1. Ağaç, örnek kümesinin tek bir düğüm ile temsil edilmesiyle başlar.
2. Eğer örnek kümesindeki tüm örnek değerleri aynı sınıfa (yani hedef değere) gidiyorsa, bu düğüm bir yaprak haline gelir ve bu hedef değerinin adını alır. Ağaç, yaprak değerinden sonra o yönde daha fazla uzamaz.
3. Diğer koşulda, algoritma nitelik-listesinden bilgi kazancı en yüksek olan niteliği seçer ve bu nitelik, test-niteliği haline gelir.
4. Test niteliğine ait olan tüm olası değerler ve bunları oluşturan örnek kümesi değerleri, bir dal şeklinde ağaca eklenir. Burada önemli olan nokta, bu değerlerin sürekli değişken şeklinde olmamasıdır. Ağacın boyutlarının kontrol edilebilir olabilmesi için, sürekli değişkenler kategorik değişken haline çevrilmiş olmalıdır.
5. Algoritma bundan sonraki aşamada, yinelemeli olarak her örnek kümesi değeri için yaprak düğüme ulaşana dek program yeni örnek kümeleri ve yeni nitelik listesi ile kendi içinde döndürülür.
6. Yinelemeli döndürme sadece aşağıdaki koşullarda durur :
    a. Tüm örnek küme hedef değerleri aynı sınıfa aitse
    b. Örneklemenin devam edebileceği bir nitelik lkistesi kalmamışsa
    c. Örnekleyecek değerler kümesi kalmamışsa.

Algoritmayı daha iyi anlayabilmek için aşağıdaki örnekten faydalanılabilir :

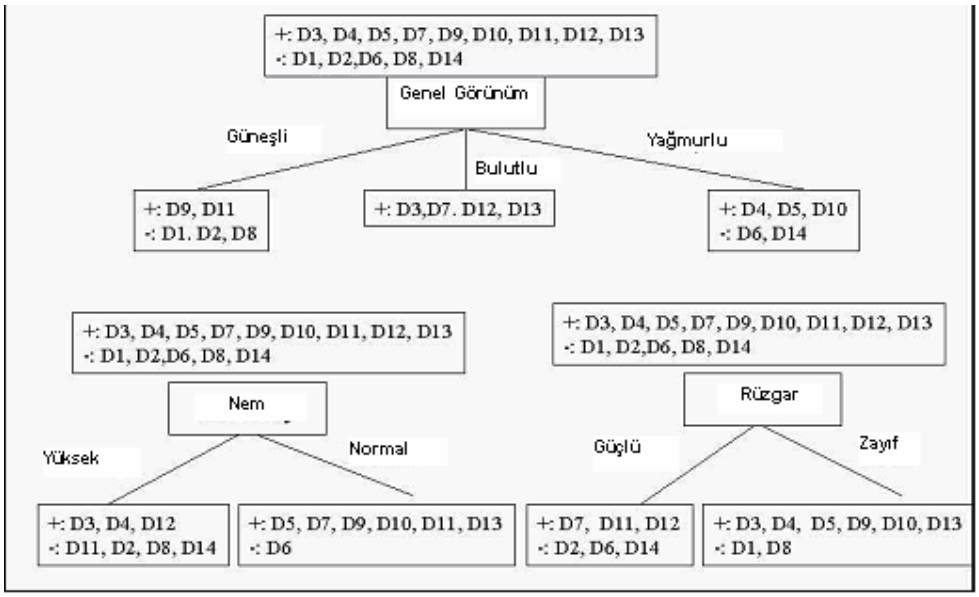
Elimizde 14 adet örnekten ve 3 adet nitelikten oluşan bir veri kümesi olsun. Bu kümenin değerleri Şekil 3.1'de görülebilir.

Havanın durum kombinasyonlarına göre, hedef değeri olan "tenis oynama/oynamama" durumu en sağdaki kolonda listelenmiştir.

Örnek	Nitelikler			Hedef Tenis Oynama
	Genel Görünüm	Nem	Rüzgar	
D1	Güneşli	Yüksek	Zayıf	Hayır
D2	Güneşli	Yüksek	Güçlü	Hayır
D3	Bulutlu	Yüksek	Zayıf	Evet
D4	Yağmurlu	Yüksek	Zayıf	Evet
D5	Yağmurlu	Normal	Zayıf	Evet
D6	Yağmurlu	Normal	Güçlü	Hayır
D7	Bulutlu	Normal	Güçlü	Evet
D8	Güneşli	Yüksek	Zayıf	Hayır
D9	Güneşli	Normal	Zayıf	Evet
D10	Yağmurlu	Normal	Zayıf	Evet
D11	Güneşli	Normal	Güçlü	Evet
D12	Bulutlu	Yüksek	Güçlü	Evet
D13	Bulutlu	Normal	Zayıf	Evet
D14	Yağmurlu	Yüksek	Güçlü	Hayır

Şekil 3.1 : Örnek bir veri kümesi

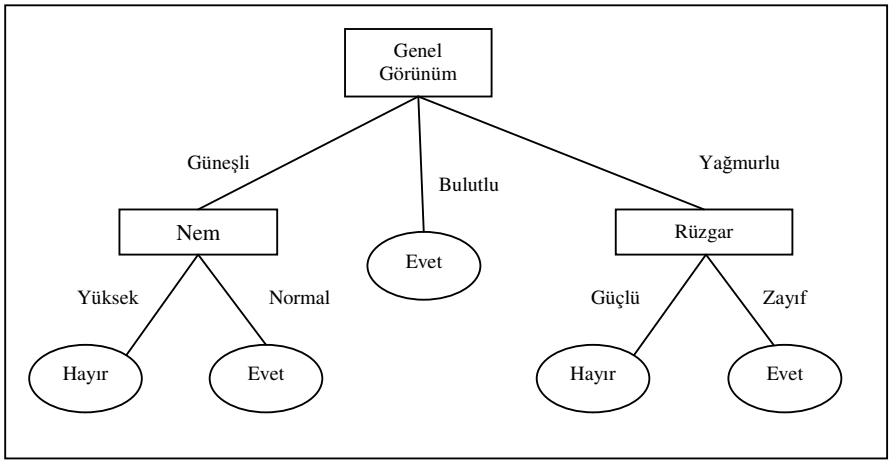
Karar ağacı algoritması, bu 3 nitelikten en yüksek bilgi kazancı olan değeri "Genel Görünüm" olarak belirler. Bu sebeple, ilkönce tüm örnek kümesi değerleri "Genel Görünüm" niteliği altında gruplanmak üzere etiketlenir. Bu niteliğin olası tüm değerleri için etiketin altına birer dal çizilir (örnekte bu değerler Güneşli / Bulutlu / Yağmurlu). Bu değerlere sahip olan tüm örnekler de etiketlerin altında kümeler halnde tutulur.



Şekil 3.2 : Örnek veri kümesinin karar ağacına dönüştürülmesinin birinci adımı

Şekil 3.2'de görüldüğü gibi, her bir daldan sonra, aslında aşağıda ağacın diğer alt-ağaçları oluşmaktadır. Örneğin, "Genel Görünüm" niteliği "Bulutlu" değerini

aldığında, tüm örnek değerler aynı hedef değerine gittiklerinden (Tenis oynanır : Evet) , "Bulutlu" değerinin altında bir alt-ağaç oluşmamış, artık bu değer bir yaprak haline gelmiştir. Ancak "Güneşli" değeri için henüz örnekleme kümesinin tümü aynı değere gitmediğinden ve nitelik listesinin de tüm elemanları kullanılmamış olduğundan, ağaç oluşturma işlemi devam edecek demektir. Bu durumda, "Genel Görünüm"'tan sonra "Güneşli" değeri için o örnek kümesi içinden hangi niteliğin daha fazla kazançlı bilgi sağlayacağını program hesaplar, ve bunun "Nem" olduğuna karar verir. Nem alt-ağacı da oluşturulduktan sonra, görüldüğü gibi, tüm değerler aynı hedef değerine gittiğinden bu alt-ağaç da burada noktalanır. Oysa "Genel Görünüm" niteliğinde "Yağmurlu " değeri için bir alt ağaç çizilmeye çalışıldığında, bilgi kazancının maksimum olduğu nitelik bu kez Nem değil, "Rüzgar" niteliği olacaktır. Ağacın aldığı son hali aşağıdaki şekilde görülebilir :



Şekil 3.3 : Örnek veri kümesinin karar ağacına dönüştürülmesinin son adımı

Karar ağaçlarında ayırt edici olan nokta, test-niteliğin seçiminde kullanılan bilgi kazancıdır. Bu değer, ağacın ayrılış noktalarındaki verimliliği temsil eder. Algoritmada, en yüksek bilgi kazancı değeri olan nitelik, test-nitelik olarak seçilir. Bu nitelik, seçilen örnek kümesinin sınıflandırılması için gereken bilgi boyutunu minimize eder. Bu bilgi teorisi merkezli yaklaşım, bir objenin sınıflandırılmasında kullanılan test sayısını en küçük hale getirerek daha basit (ama en basit olmayan) bir ağaç yapısı ortaya çıkartır.

S, s adet veri örneğinden oluşan bir küme olsun. C_i (i=1,...,m) ise, bir niteliğe ait m adet değerin tanımlı sınıf değerleri olsun. s_i, Cisınıfında bulunan S örneklerinin sayısı

olsun. Bir örnek kümesini sınıflandırmak için kullanılan bilgi miktarının beklenen değeri, aşağıdaki formülle hesaplanır :

$$I(s_1, s_2, ..., s_m) = -\sum_{i=1}^{m} p_i \log_2(p_i) \quad \textbf{(3.1)}$$

Bu formüldeki pi, bir örneğin Ci sınıfında bulunma olasılığıdır ve si/s değerine eşittir.

A niteliğine ait v adet değer bulunsun $\{a_1, a_2..., a_v\}$. A niteliği, S'yi v adet altkümeye ayırabilir, $\{S_1, S_2...S_v\}$. Eğer, A test niteliği olarak seçildiyse, o zaman bu altkümeler, S kümesini içeren düğümün birer dalı olacak dmektir. s_{ij}, S_j altkümesinin C_i sınıfında bulunan örneklerin sayısı olsun. Entropy, ya da A altkümelerine bölünmede beklenen bilgi miktarı şu şekilde hesapla Entropy, ya da A altkümelerine bölünmede beklenen bilgi miktarı şu şekilde hesaplanır :

$$E(A) = \sum_{j=1}^{v} \frac{(s_{1j} + ... + s_{mj})}{s \cdot I(s_{1j}, ..., s_{mj})} \quad \textbf{(3.2)}$$

Entropy değeri küçüldükçe, altküme bölünmelerinin saflık derecesi artar.

$$I(s_{1j}, ..., s_{mj}) = -\sum_{i=1}^{m} p_{ij} \log_2(p_{ji}) \quad \textbf{(3.3)}$$

A niteliğinin üzerinden dallanmadan kazanılan bilgi kazancı ise :

$$Gain(A) = I(s_{1j}, ..., s_{mj}) - E(A) \quad \textbf{(3.4)}$$

Başka bir deyişle, Gain(A), A niteliğinin değeri bilindiğinde entropy'de beklenen düşüşü simgeler.

3.2. Karar Ağacının Budanması

Bir karar ağacı oluşturulduğunda, birçok dalda, öğrenme verisindeki gürültü ve kayıplardan dolayı anomaliler oluşacaktır. Ağacın budanma metodu, bu sorunu ortadan kaldırmaya yardımcı olabilir. Bu metod, tipik olarak en az güvenilir olan dalı

istatistiksel olarak hesaplayıp kaldırmaktan ibarettir ve daha hızlı ve güvenilir bir sınıflandırma ile sonuçlanır. İki adet budama yöntemi vardır.

Bunlardan birincisi, önceden-budama yöntemidir. Bu yöntemde öğrenme verisi sınıflandırılırken ağacın o dalının ileriye yönelik devam edip etmeyeceğine önceden karar verilir ve gerekiyorsa, geri kalan bölünmeden sonra geriye kalan verinin sınıflandırılması durdurularak, en fazla hedef değeri taşıyan değer yaprak yapılır. Bu yöntemde, önceden bir eşik değeri belirlenir. Bu eşik değerini aşmayan bilgi kazançlarına sahip olan nitelikler gruplandırılır. Program devam ederken bu bilgi kazancının düştüğü noktada ağacın büyümesine izin verilmeden diğer dala geçilir.

Her iki koşulda da, bu eşik değerini belirlemek işin en zor kısmıdır. Çünkü eşik değeri çok yüksek tutulursa ortaya çıkan ağaç çok fazla basit ve genel kurallardan oluşan bir ağaç olur. Eşik değeri çok düşük tutulursa ise, ağacın sınıflandırması çok özele inebilir ve test verisi üzerinde doğru sonuçlar ortaya çıkmayabilir.

İkinci yöntem, sonradan-budama yöntemidir. Tamamen büyümüş bir ağaç üzerinde uygulanır. Tüm dalların çıkardığı kurallar denenerek, bunlardan en fazla hata oranını oluşturan dal budanır. Böylece ortaya daha basit bir ağaç yapısı çıkartılabilir [1].

Alternatif olarak, hem önceden hem de sonradan budama yöntemi birleştirilerek yeni bir yöntem olarak kullanılabilir. Sonradan-budama yöntemi önceden-budama'ya göre çok daha fazla hesaplama gerektirir ancak ortaya daha güvenilir bir ağaç çıkartır.

4. UYGULAMA

4.1. Veri Kümesinin Tanımlanması

Müşteri kaybedilmesinin ölçümlenmesinde zor olan taraf, müşterinin hangi koşullarda kaybedildiğinin ortaya çıkarılmasıdır. Çünkü hesaba katılması gereken bir çok nitelik olabilir.

Güçlü bir model oluşturabilmek için işin amaçlarına ve hedeflerine ters düşmeyecek bir "müşteri kaybı tanımı" ortaya konmalıdır.

Bu tezde kullanılan veriler için müşteri kaybı tanımı, banka tarafından aşağıdaki gibi belirlenmiştir :

a. Açık hesap sayısı 0 olanlar veya

b. Son hesap hareketi tarihi 6 ay önce olanlar.

Şekil 4.1'de, bu analizi yaparken, banka müşterilerine ait hangi niteliklerin kullanıldığı sıralaması bulunmaktadır.

Burada belirtilen niteliklerden, CIF_ID, müşteri numarasıdır, ve müşteri güvenliği sebebiyle bu tezde değiştirilerek kullanılmıştır.

En sonda belirtilen CHURN_FLAG ise, müşterinin yukarıdaki tanıma göre kaybedilip edilmediği bilgisini içerir. Eğer müşteri kaybedilmişse bu bayrak "T", kaybedilmemişse "F" değerini alır. Niteliklerin yanlarında kısaltmalar bulunmaktadır.

Bankadan, tabloda belirtilen nitelik bilgilerine sahip 30,000 adet kayıt alınmıştır. Bu kayıtların 10,000 tanesi kaybedilmiş, 20,000 tanesi ise kaybedilmemiş müşterilerden oluşmaktadır.

Analiz için kullanılan yöntem, karar ağaçlarıdır.

	Nitelik Adı	Açıklama
1	CIF_ID	Müşteri numarası
2	NUMBER_OF_ALL_ACCOUNTS (allac)	Tüm hesapların sayısı
3	NUMBER_OF_OPEN_ACCOUNTS (openacc)	Tüm açık hesapların sayısı
4	LAST_TXN_DATE_OF_ALL_ACCOUNTS (lasttxnall)	Tüm hesaplarındaki son hareket tarihi
5	MIN_FIRST_OPEN_DATE_OF_ALL_ACCOUNTS (mindateall)	Tüm hesaplarının minimum açılış tarihi
6	MIN_FIRST_OPEN_DATE_OF_OPEN_ACCOUNTS (mindateopen)	Tüm açık hesaplarının minimum açılış tarihi
7	MAX_FIRST_OPEN_DATE_OF_OPEN_ACCOUNTS (maxdateopen)	Tüm açık hesaplarının maksimum açılış tarihi
8	MEAN_12 (mean12)	12 aylık harcama ortalaması
9	STD_DEV_12 (stdev12)	12 aylık harcamanın standart sapması
10	LAST_03_MONTH_SUM_ALL (last3sumall)	Son 3 aylık tüm harcamaların toplamı (Nakit çekimler dahil)
11	LAST_06_MONTH_SUM_ALL (last6sumall)	Son 6 aylık tüm harcamaların toplamı (Nakit çekimler dahil)
12	LAST_03_MONTH_SUM_ALISVERIS (last3alver)	Son 3 aylık tüm alışveriş harcamasının toplamı
13	LAST_06_MONTH_SUM_ALISVERIS (last6alver)	Son 6 aylık tüm alışveriş harcamasının toplamı
14	LAST_03_MONTH_SUM_TEXTILE (last3textile)	Son 3 aylık tekstil alışverişi harcamasının toplamı
15	LAST_06_MONTH_SUM_TEXTILE (last6textile)	Son 6 aylık tekstil alışverişi harcamasının toplamı
16	LAST_03_MONTH_SUM_FOOD (last3food)	Son 3 aylık yiyecek alışverişi harcamasının toplamı
17	LAST_06_MONTH_SUM_FOOD (last6food)	Son 6 aylık yiyecek alışverişi harcamasının toplamı
18	LAST_03_MONTH_SUM_FUEL (last3fuel)	Son 3 aylık benzin alışverişi harcamasının toplamı
19	LAST_06_MONTH_SUM_FUEL (last6fuel)	Son 6 aylık benzin alışverişi harcamasının toplamı
20	LAST_TXN_DATE_OF_PFT (lasttxnpft)	Son PFT tarihi
21	LAST_TXN_DATE_OF_VFT (lasttxvft)	Spn VFT tarihi
22	INTERNET_LAST_TXN_DATE (lasttxnint)	İnternetten yapılan en son işlem tarihi
23	TOPLANAN_PUAN_ORANI_SON_3_EKSTRE_DONEMI (last3puan)	Son 3 ekstre dönemine, toplanan worldpuan oranı
24	TOPLANAN_PUAN_ORANI_SON_6_EKSTRE_DONEMI (last6puan)	Son 6 ekstre dönemine, toplanan worldpuan oranı
25	LAST6_WORLDMERCH_WORLDCARD_SUM_RATIO (last6worldmerch)	Son 6 aylık Worldmerch harcamasının Worldcard harcamasına oranı
26	SON_CEPTEN_VADAA_KONTOR_YUKLEME_TARIHI (lastcepvada)	Kredi kartından cep telefonuna en son kontör gönderme tarihi
27	KAMP_KAZANILAN_KONTOR_ADEDI_SON_1_YIL (kontor)	Son 1 yılda kapmanyalardan kazanılan kontör adedi
28	MAILORDER_COUNT_12 (mailordercount)	12 ayda yapılan mail siparişi sayısı
29	TELEWEB_MEMBERSHIP_FLAG (teleweb)	Teleweb (internet bankacılığı) üyeliği var mı bayrağı
30	CHURN_FLAG	Müşterinin kaybedilmiş olma bayrağı

Şekil 4.1 : Veri kümesinde kullanılan nitelikler ve açıklamaları

4.2. Veri Temizleme ve Dönüştürme

Aşağıdaki şekilde, 30,00 kayıtlık verinin bir kısmı örnek olarak gösterilmektedir :

```
35440608,1,1,2002,2002,2002,8,0-30,0-60,0,0-200,0,0-200,0,0,0,0,0,0,2005,null,null,0.06,0.85>,null,null,null,0,1,T
35450230,1,1,2002,2002,2002,11,100-200,140-208,500-900,500-900,200-500,500-900,0,0,75-175,50-150,0,0,2004,null,0.01,0.01
35460243,2,1,1996,1996,1996,7,0-30,0-60,0,0-200,0,0-200,0,65-230,0,0,0,0,2005,null,null,0.01,0.85>,null,null,null,0,0,T
35467519,2,1,1996,1996,1996,7,0-30,0-60,0,0-200,0,0-200,0,0-65,0,0,0,0,2003,null,null,0.01,0,null,null,null,0,0,T
35483601,1,1,2004,2004,2004,12,330-500,500-700,2700-5000,5000-15000,500-900,1400-2000,65-230,230-600,175-350,250-500,100
35506636,2,1,1996,1996,1996,7,30-100,140-208,0,500-900,0,500-900,0,0,0,0,0,0,2005,null,null,0.01,0.85>,null,null,null,0,
35523347,1,1,2004,2004,2004,7,0-30,0-60,2700-5000,2700-5000,0,0-200,0,0,0,0,0,0,null,null,null,0.01,0,null,null,null,0,1
35524622,1,1,1996,1996,1996,11,0-30,0-60,0-200,0-200,0-200,0-200,0,0,0,50-150,0,0-70,null,null,0.01,0.01,0.35-0.5,null,2
35529909,1,1,1996,1996,1996,12,200-330,208-350,900-1400,2000-2700,900-1400,2000-2700,0,0,0-75,50-150,0-100,70-150,2005,r
35541299,2,2,1996,1996,1998,10,100-200,208-350,0-200,2000-2700,0-200,200-500,0,230-600,0,0,0,0,2005,null,null,null,0-0.2
35553947,2,1,1990,2003,2003,8,30-100,0-60,1400-2000,1400-2000,0,0-200,0,0,0,0,0,0,2005,null,0,null,0.5-0.65,null,04.08.2
35576931,1,1,2002,2002,2002,12,100-200,60-140,500-900,900-1400,500-900,900-1400,230-600,230-600,175-350,500-850,100-250,
35577237,1,1,1997,1997,1997,11,100-200,208-350,500-900,2000-2700,0-200,900-1400,65-230,65-230,0,150-250,0,0,2005,null,0,
```

Şekil 4.2 : Veri kümesinden örnek bir görüntü

Veriyi dönüştürmek için aşağıdakiler yapıldı :

LAST_TXN_DATE_OF_ALL_ACCOUNTS : Bu nitelik sadece 2005 yılına ait olduğundan, farklılık oluşturması açısından yalnızca ay kısmı kullanıldı.

MIN_FIRST_OPEN_DATE_OF_ALL_ACCOUNTS,
MIN_FIRST_OPEN_DATE_OF_OPEN_ACCOUNTS,
MAX_FIRST_OPEN_DATE_OF_OPEN_ACCOUNTS,
LAST_TXN_DATE_OF_PFT, LAST_TXN_DATE_OF_VFT : Bu nitelikler için tarihlerin sadece yıl bilgisi kullanıldı.

MEAN_12, STD_DEV_12, LAST_03_MONTH_SUM_ALL,
LAST_06_MONTH_SUM_ALL, LAST_03_MONTH_SUM_ALISVERIS,
LAST_06_MONTH_SUM_ALISVERIS, LAST_03_MONTH_SUM_TEXTILE,
LAST_06_MONTH_SUM_TEXTILE, LAST_03_MONTH_SUM_FOOD,
LAST_06_MONTH_SUM_FOOD, LAST_03_MONTH_SUM_FUEL,
LAST_06_MONTH_SUM_FUEL: Bu niteliklerinde frekanslarına bakılarak eşit genişlikli histogram yöntemiyle aralıklar belirlendi. Tüm veriler bu aralıklara oturtulacak şekilde dönüştürüldü.Eksik olanlara "0" değeri atıldı.

TOPLANAN_PUAN_ORANI_SON_3_EKSTRE_DONEMI,
TOPLANAN_PUAN_ORANI_SON_3_EKSTRE_DONEMI :Bu niteliklerde boş olan kayıtlar için "null" değeri atandı.

INTERNET_LAST_TXN_DATE,
SON_CEPTEN_VADAA_KONTOR_YUKLEME_TARIHI : Bu nitelikler sonradan veri ambarına eklendiği için eksik kayıtları çok fazlaydı. Dönüşüm için tarihlerin yıl/ay bilgileri kullanıldı. Eksik veriler için ise "null" değeri atıldı.

KAMP_KAZANILAN_KONTOR_ADEDI_SON_1_YIL: Niteliğe ait kayıtlarda dönüştürme işlemi yapılmasına gerek yoktu, ancak çok fazla eksik veri içerdiğinden boş bilgilerin yerine "null" değeri atıldı.

LAST6_WORLDMERCH_WORLDCARD_SUM_RATIO: Nitelikte önce histogram yöntemiyle aralıklar belirlenerek veri üzerinde dönüşüm yapıldı. Sonra boş bilgiler için "null" değeri atandı.

4.3. Karar Ağacı Algoritmasının Veri Üzerinde Uygulanması

Bu algoritma, öncelikle 198 kayıt üzerinde uygulanmıştır. Bu kayıtların yarısı kaybedilmiş, yarısı kaybedilmemiş müşterilerin bilgilerinden oluşmaktadır. Algoritma çalıştırılmadan önce ekrandan bir eşik değeri istenir. Bu eşik değerine göre hangi niteliklerin ağacın yapısına dahil edilip edilmeyeceğine karar verilir. Eşik

değerinin belirlenmesi için ise ilkönce tüm niteliklerin 198 adet kayıttaki bilgi kazanç değerleri hesaplanmıştır. Çıkan sonuç Şekil 4.3'de gösterilmiştir.

Açık olan hesap sayısının kazancının en düşük çıkmasının sebebi, entropy değerinin en yüksek olmasıdır. Çünkü müşterinin kaybedilmesi kriterleri belirlenirken bu değer kullanılmıştır.

Karar ağacı algoritması, daha sonra farklı eşik değerleri için çalıştırılmıştır. Eşik değeri 0.37 iken ortaya çıkan karar ağacı Şekil 4.4'de gösterilmiştir.

Nitelik Adı	Bilgi Kazancı
LAST_TXN_DATE_OF_ALL_ACCOUNTS	0.462641
LAST_03_MONTH_SUM_ALISVERIS	0.378131
LAST_03_MONTH_SUM_FOOD	0.36715
LAST_03_MONTH_SUM_ALL	0.331354
LAST_06_MONTH_SUM_ALISVERIS	0.22157
LAST_03_MONTH_SUM_TEXTILE	0.220138
LAST_06_MONTH_SUM_TEXTILE	0.215547
LAST_06_MONTH_SUM_FOOD	0.214829
LAST_03_MONTH_SUM_FUEL	0.209738
LAST_06_MONTH_SUM_ALL	0.194534
MAILORDER_COUNT_12	0.156768
INTERNET_LAST_TXN_DATE	0.153418
MEAN_12	0.145724
TOPLANAN_PUAN_ORANI_SON_3_EKSTRE_DONEMI	0.144167
LAST_TXN_DATE_OF_PFT	0.13858
MAX_FIRST_OPEN_DATE_OF_OPEN_ACCOUNTS	0.12243
STD_DEV_12	0.119614
LAST6_WORLDMERCH_WORLDCARD_SUM_RATIO	0.114724
MIN_FIRST_OPEN_DATE_OF_OPEN_ACCOUNTS	0.108382
SON_CEPTEN_VADAA_KONTOR_YUKLEME_TARIHI	0.098418
MIN_FIRST_OPEN_DATE_OF_ALL_ACCOUNTS	0.094296
LAST_06_MONTH_SUM_FUEL	0.084871
TELEWEB_MEMBERSHIP_FLAG	0.064938
KAMP_KAZANILAN_KONTOR_ADEDI_SON_1_YIL	0.064471
TOPLANAN_PUAN_ORANI_SON_6_EKSTRE_DONEMI	0.052924
LAST_TXN_DATE_OF_VFT	0.017453
NUMBER_OF_ALL_ACCOUNTS	0.015589
NUMBER_OF_OPEN_ACCOUNTS	0.008022

Şekil 4.3 : İlk analizde tüm niteliklere ait bilgi kazançları

Eşik değeri 0.37 iken, algoritma ağacı oluştururken sadece bu eşik değerinin üzerindeki nitelikleri hesaba katmıştır ve ortaya göreceli olarak küçük bir ağaç çıkmıştır. Ancak eşik değeri küçültülürse ve 0.1'e çekilirse, çıkan karar ağacı çok büyük ve detaylı olmaktadır. Şekil 4.3'de, bu çıkan ağacın bir bölümü görülebilir.

```
Hangi esik degerinin altindaki nitelikler alinmasin? :0.37

lasttxnall:0.462641
last3alver:0.378131

 lasttxnall=12 last3alver=1400-2000 Nd=F
               last3alver=0 Nd=X
               last3alver=0-200 Nd=F
               last3alver=2700-5000 Nd=F
               last3alver=5000-15000 Nd=F
               last3alver=200-500 Nd=F
               last3alver=2000-2700 Nd=F
               last3alver=500-900 Nd=F
               last3alver=900-1400 Nd=F
               last3alver=15000> Nd=F

 lasttxnall=8 Nd=T
 lasttxnall=9 Nd=T
 lasttxnall=7 Nd=T
 lasttxnall=10 last3alver=0-200 Nd=T
               last3alver=200-500 Nd=X
               last3alver=900-1400 Nd=T

 lasttxnall=11 last3alver=1400-2000 Nd=F
               last3alver=0-200 Nd=T
               last3alver=2700-5000 Nd=F
               last3alver=200-500 Nd=F
               last3alver=2000-2700 Nd=F
               last3alver=500-900 Nd=T
               last3alver=900-1400 Nd=T
               last3alver=15000> Nd=T

 lasttxnall=6 Nd=T
```

Şekil 4.4 : İlk analizde eşik değeri 0.37 iken çıkan karar ağacından bir görüntü:

```
Hangi esik degerinin altindaki nitelikler alinmasin? :0.1

mindateopen:0.108382
maxdateopen:0.122430
lasttxnall:0.462641
mean12:0.145724
stdev12:0.119614
last3sumall:0.331354
last6sumall:0.194534
last3alver:0.378131
last6alver:0.221570
last3fuel:0.209738
last3food:0.367150
last6food:0.214829
last3textile:0.220138
last6textile:0.215547
lasttxnpft:0.138580
last6worldmerch:0.114724
lasttxnint:0.153418
mailordercount:0.156768

 lasttxnall=12 mindateopen=1998 maxdateopen=1998 mean12=800-2000 Nd=T
                                                 mean12=200-330 Nd=F
                                                 mean12=330-500 Nd=T

               mindateopen=2004 maxdateopen=2004 mean12=200-330 Nd=F
                                                 mean12=0-30 stdev12=60-140 Nd=F

                                                             stdev12=0-60 Nd=T
                                                 mean12=330-500 Nd=F
               mindateopen=1999 Nd=F
               mindateopen=2003 maxdateopen=2003 mean12=100-200 stdev12=140-208 Nd=F
                                                             stdev12=208-350 Nd=T
                                                 mean12=330-500 Nd=F
                                                 mean12=2000> Nd=F
               mindateopen=2001 maxdateopen=2001 mean12=0-30 Nd=F
                                                 mean12=330-500 Nd=T
                                                 mean12=500-800 Nd=F
               mindateopen=2005 maxdateopen=2005 mean12=30-100 stdev12=140-208 Nd=F
                                                               stdev12=60-140 Nd=T
                                                 mean12=100-200 Nd=F
                                                 mean12=330-500 Nd=F
               mindateopen=1995 maxdateopen=1995 mean12=200-330 stdev12=140-208 last3sumall=500-900 last6sumall=900-14
                                                                                                    last6sumall=2000-2
```

Şekil 4.5 : İlk analizde eşik değeri 0.3 iken çıkan karar ağacından bir görüntü

Nd=X olarak görülen değer, hedef değerin True ya da False olması olasılığının birbirine eşit olduğu durumlarda ortaya çıkar. Bu karar ağacındaki her bir satır, artık bir kuraldır. Bu kurallar, daha sonra test verisi üzerinde denenerek ne kadar doğru sonuç verdiği ortaya çıkartılır. Nd=X olan bir ağaç dalı ya da kural, test algoritmasına dahil edilmemiştir, çünkü ortaya bilinen bir sonuç çıkarmamaktadır. Farklı eşik değerlerine göre çıkan sonuçlar aşağıdaki tablolarda bulunabilir :

Tablo 4.1 : İlk analizde eşik değeri 0.1 iken çıkan doğru sayısı ve kurala uyma sayıları ile oranları

Eşik = 0.1	Doğru Sayısı	Kurala Uyma	Doğru Oran	Kurala Uyma Oran
Öğrenilen Veri Kümesi İçin	196	198	0.9899	1
198 Kayıtlık Test Verisi İçin	77	100	0.3889	0.505
2000 Kayıtlık Test Verisi İçin	807	1079	0.4035	0.5395

Tablo 4.2 : İlk analizde eşik değeri 0.15 iken çıkan doğru sayısı ve kurala uyma sayıları ile oranları

Eşik = 0.15	Doğru Sayısı	Kurala Uyma	Doğru Oran	Kurala Uyma Oran
Öğrenilen Veri Kümesi İçin	181	181	0.9142	0.9142
198 Kayıtlık Test Verisi İçin	119	150	0.601	0.7575
2000 Kayıtlık Test Verisi İçin	1207	1516	0.6035	0.758

Tablo 4.3 : İlk analizde eşik değeri 0.2 iken çıkan doğru sayısı ve kurala uyma sayıları ile oranları

Eşik = 0.2	Doğru Sayısı	Kurala Uyma	Doğru Oran	Kurala Uyma Oran
Öğrenilen Veri Kümesi İçin	188	188	0.9494	0.9494
198 Kayıtlık Test Verisi İçin	119	156	0.601	0.7878
2000 Kayıtlık Test Verisi İçin	1187	1535	0.5935	0.7675

Tablo 4.4 : İlk analizde eşik değeri 0.3 iken çıkan doğru sayısı ve kurala uyma sayıları ile oranları

Eşik = 0.3	Doğru Sayısı	Kurala Uyma	Doğru Oran	Kurala Uyma Oran
Öğrenilen Veri Kümesi İçin	171	182	0.8636	0.9191
198 Kayıtlık Test Verisi İçin	122	152	0.6161	0.7676
2000 Kayıtlık Test Verisi İçin	1307	1632	0.6535	0.816

Tablo 4.5 : İlk analizde eşik değeri 0.36 iken çıkan doğru sayısı ve kurala uyma sayıları ile oranları

Eşik = 0.36	Doğru Sayısı	Kurala Uyma	Doğru Oran	Kurala Uyma Oran
Öğrenilen Veri Kümesi İçin	173	186	0.8737	0.9393
198 Kayıtlık Test Verisi İçin	131	165	0.6616	0.8333
2000 Kayıtlık Test Verisi İçin	1376	1747	0.688	0.8735

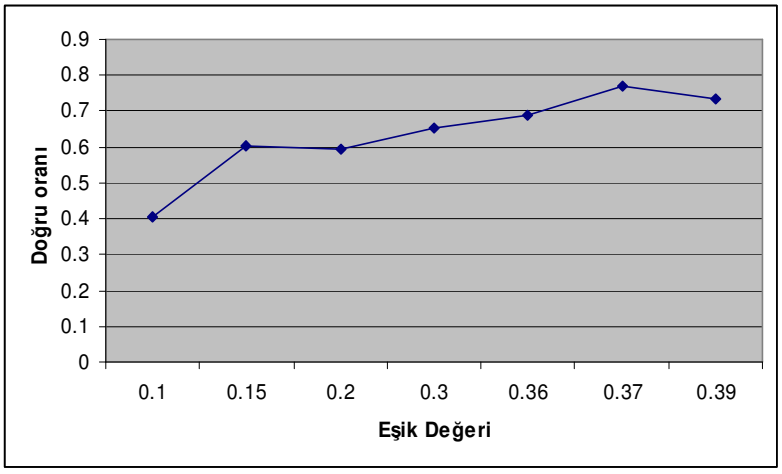
Tablo 4.6 : İlk analizde eşik değeri 0.37 iken çıkan doğru sayısı ve kurala uyma sayıları ile oranları

Eşik = 0.37	Doğru Sayısı	Kurala Uyma	Doğru Oran	Kurala Uyma Oran
Öğrenilen Veri Kümesi İçin	166	194	0.8383	0.9797
198 Kayıtlık Test Verisi İçin	153	189	0.7727	0.9545
2000 Kayıtlık Test Verisi İçin	1542	1913	0.771	0.9565

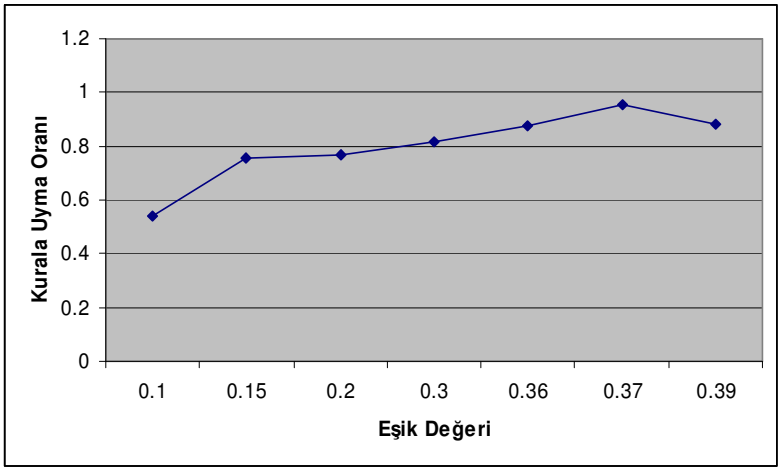
Tablo 4.7 : İlk analizde eşik değeri 0.39 iken çıkan doğru sayısı ve kurala uyma sayıları ile oranları

Eşik = 0.39	Doğru Sayısı	Kurala Uyma	Doğru Oran	Kurala Uyma Oran
Öğrenilen Veri Kümesi İçin	153	178	0.7727	0.8989
198 Kayıtlık Test Verisi İçin	143	171	0.7222	0.8636
2000 Kayıtlık Test Verisi İçin	1471	1763	0.7355	0.8815

2000 kayıtlık test verisi üzerinde yapılan çalışmanın sonuçları grafiklere dökülecek olursa, eşik değeri yükseldikçe ortaya çıkartılıp test edilen kurallara uygunluk ve doğru sonuçlara ulaşma oranı genel bir artış göstermiştir. Bu iki grafikteki çizgilerin üstüste geldiğinde birbirine benzer olması, bize sınıflandırmayı ne kadar doğru yapıp yapmadığımızı, daha doğrusu, tahminlemenin ne kadar iyi olduğu bilgisini verir. Eğer, doğru oranı çizgisi, kurala uyma çizgisinden farklılıklar gösteriyorsa, bu, test edilen kurala tam olarak uyan bir test verisi bulunduğu halde, bu müşterinin kayıp bilgisinin (T ya da F) doğru tahminleme yapılmadığını gösterecektir. Aşağıdaki grafiklerde ise kurala uygunluk ile doğruluk oranının yaklaşık olarak aynı artışlar veya düşüşleri gösterdiği gözlenmektedir. Bu da, karar ağacından faydalanılarak bulunan kuralların aslında bizi doğru hedef değerlerine götürdüğünü göstermektedir.

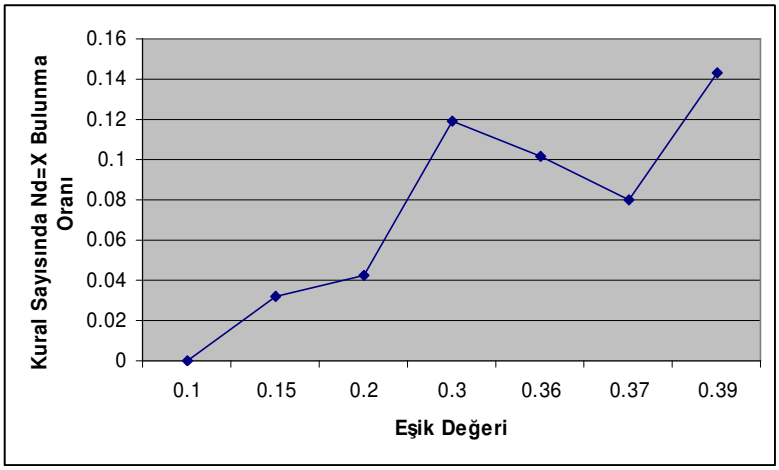


Şekil 4.6 : İlk analizde eşik değeri değişiminin doğru bulunan hedef değeri oranına etkisi



Şekil 4.7 : İlk analizde eşik değeri değişiminin kurala uyma oranına etkisi

Nd=X ile biten ağaç dallarının sayısının toplam kural sayısına oranı ile eşik değeri karşılaştırıldığında ise şöyle bir grafik ortaya çıkmaktadır :

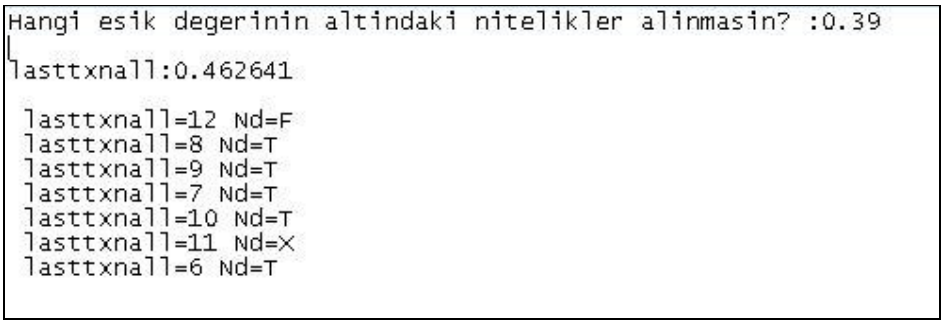


Şekil 4.8 : İlk analizde eşik değeri değişiminin kural sayısında Nd=X bulunma oranına etkisi

Grafiklerden ortaya çıkan sonuç, eşik değeri arttıkça, ortaya çıkan kural sayısı azalır ve daha basit bir ağaç yapısı bulunur. Dolayısıyla, test verisi üzerinde eşik değeri büyük olan karar ağacı kuralları uygulandığında daha genel bir kural tablosuna oturan, bu sebeple de daha doğru sonuçlar veren bir ağacın ortaya çıktığı düşünülebilir.Ancak, Nd=X sayısına bakıldığında görülecektir ki, ağaç basit olmasına rağmen, ortaya daha az güvenli ve daha belirsiz bir sınıflandırma çıkar.

Örneğin, en büyük eşik değeri olan 0.39 eşiği ile ortaya çıkan ağaca bakıldığında 6 kuraldan oluşan bir ağaç yapısı çıkar. Ancak, bunlardan 1 tanesi Nd=X ile bitmektedir. Yani, karar ağacı, son hesap hareketi tarihi 10.ayda olan bir kişinin kaydı için tahminleme yapmak istediğinde, hedeflerin yarısı T, yarısı F değerlerini aldığından, bir tahmin yapamayacaktır. Kurallar test edildiğinde, Nd=X ile biten kurallar test kural tablosuna dahil edilmediğinden, bu aya ait bir bilgiye hiçbir şekilde ulaşılamamıştır. Oysa, eşik değeri biraz küçültülerek 0.37 olarak seçilirse ve ağaca bir tane daha nitelik eklenirse, en azından 10. ay içinde son hesap hareketi olanlar için daha detayda, son 3 aylık alışveriş harcaması toplamının 0-200 veya 900-1400 değerleri arasında değişenleri için hedef değeri tahminlemesi yapılabilmektedir.

Yukarıda bahsedilen avantajlar ve dezavantajlar, eşik değerinin seçiminde önemlidir ve analizi yapılan veriden ne öğrenilmek istendiğine bağlı olarak biri ya da öbürü seçilmelidir.



Şekil 4.9 : İlk analizde eşik değeri 0.39 iken orataya çıkan karar ağacı

Eğer amaç sadece doğru tahminleme yapmaksa eşik değeri büyütülerek daha fazla doğru sonuca ulaşılması sağlanabilir. Amaç eldeki veriden analizler yapıp sonuçlar çıkarmaksa ise, eşik değeri daha küçük kullanılmalı ve daha detaylı bir karar ağacı incelenmelidir.

198 kayıtlık öğrenme verisi, ikinci aşamada , daha spesifik ve doğru kurallar elde edilebilmesi için 10,000 kayıt ile değiştirilmiştir. Yeni analizde bilgi kazançları sıralaması Şekil 4.10'daki gibi olmuştur.

Bu kazançlar, 198 kayıtlık verideki kazanç sıralamasına benzer sonuçlar çıkarmıştır. Tahmin edildiği gibi, yine açık hesap sayısının bilgi kazancı en düşük ikinci, en yüksek kazanç sağlayan nitelik ise son hesap hareketi tarihidir.

Farklı olan taraf, son 3 ekstre döneminde toplanan puan oranı niteliğinin oldukça yüksek bir kazanç sağladığıdır. Ancak bu niteliğe ait değerler genel olarak boş olduğundan, bu boş değerlerin yerine "null" sabit değeri atıldığından, algoritma null değerini boş olarak algılayamamış ve kazancın yüksek olduğunu varsaymıştır.

Bu sebeple, bu değer bundan sonraki karar ağacı oluşturma aşamasında, hesaba katılmayacaktır. Programda bunun için özel bir kontrol konmuştur. Diğer farklılıklar ise beklenen değerlerden çok farklı çıkmamıştır.

Ağaç oluşturulurken öncelikle, tek niteliğin gözönüne alınabilmesi adına, eşik değeri 0.3 seçilmiştir. Yeni ağaca bakıldığında, ilk analizde ortaya çıkan 6 dallı ağaca çok benzer bir sonuç ortaya çıkmıştır.

Ancak bu kez farklı olan, son hesap hareketi tarihi 10.ay olan kuralın, çoğunluğu sağlayabilecek hedef değeri bu kez elde edebilmesi ve Nd=X çıkarmamasıdır. Bu da bize, öğrenme veri kümesinin büyüklüğü arttıkça çıkan karar ağacının yapısının daha kesin sonuçlar verebildiğini göstermektedir.

Nitelik Adı	Bilgi Kazancı
LAST_TXN_DATE_OF_ALL_ACCOUNTS	0.353778
MIN_FIRST_OPEN_DATE_OF_ALL_ACCOUNTS	0.277901
LAST_03_MONTH_SUM_ALISVERIS	0.246984
TOPLANAN_PUAN_ORANI_SON_3_EKSTRE_DONEMI	0.224694
LAST_03_MONTH_SUM_ALL	0.206283
MIN_FIRST_OPEN_DATE_OF_OPEN_ACCOUNTS	0.191385
MAX_FIRST_OPEN_DATE_OF_OPEN_ACCOUNTS	0.168056
LAST_03_MONTH_SUM_FOOD	0.150632
LAST_06_MONTH_SUM_ALISVERIS	0.111305
LAST_03_MONTH_SUM_TEXTILE	0.108019
SON_CEPTEN_VADAA_KONTOR_YUKLEME_TARIHI	0.098795
LAST_03_MONTH_SUM_FUEL	0.088333
LAST_06_MONTH_SUM_TEXTILE	0.083161
LAST_06_MONTH_SUM_ALL	0.081373
LAST_06_MONTH_SUM_FOOD	0.081206
TOPLANAN_PUAN_ORANI_SON_6_EKSTRE_DONEMI	0.071552
LAST6_WORLDMERCH_WORLDCARD_SUM_RATIO	0.06832
MEAN_12	0.061738
LAST_06_MONTH_SUM_FUEL	0.045891
LAST_TXN_DATE_OF_PFT	0.038418
STD_DEV_12	0.034494
LAST_TXN_DATE_OF_ALL_ACCOUNTS	0.026686
MAILORDER_COUNT_12	0.005186
KAMP_KAZANILAN_KONTOR_ADEDI_SON_1_YIL	0.004647
LAST_TXN_DATE_OF_VFT	0.00389
TELEWEB_MEMBERSHIP_FLAG	0.00018
NUMBER_OF_OPEN_ACCOUNTS	0.000164
INTERNET_LAST_TXN_DATE	0

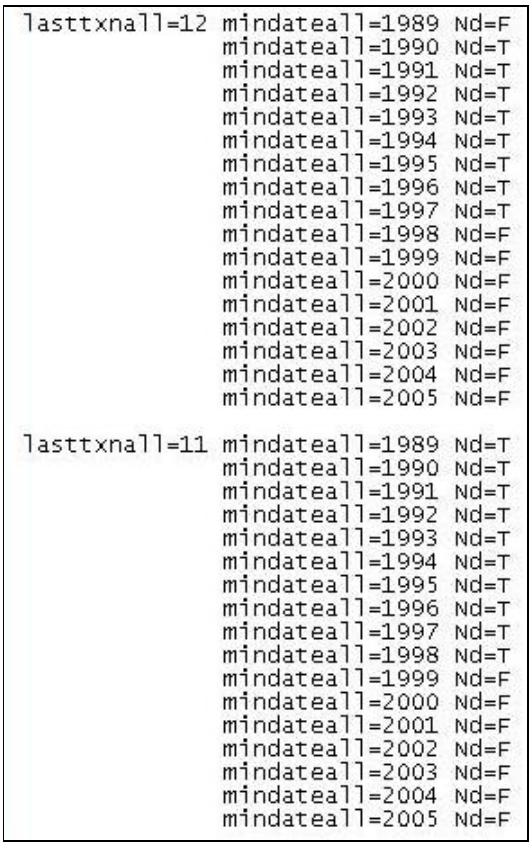
Şekil 4.10 : İkinci analizde tüm niteliklere ait bilgi kazançları

```
Hangi esik degerinin altindaki nitelikler alinmasin? :0.3

lasttxnall:0.353778

 lasttxnall=12 Nd=F
 lasttxnall=8 Nd=T
 lasttxnall=9 Nd=T
 lasttxnall=11 Nd=F
 lasttxnall=7 Nd=T
 lasttxnall=10 Nd=T
 lasttxnall=6 Nd=T
```

Şekil 4.11 : İkinci analizde eşik değeri 0.3 iken oratya çıkan karar ağacı

Asıl eşik değeri saptanıp, bulunacak olan kurallara geçmeden önce, karar ağacı algoritması bir kez de 2 nitelik alabilecek şekilde, 0.25 eşik değeri için çalıştırılmıştır. Çıkan ağaç yapısı Şekil 4.12 ve 4.13'de görülebilir.

11. ve 12. aylarda son hesap hareketi bulunan bir kişinin tüm hesaplarının minimum açılma tarihi 2005 yılına yaklaştıkça, bu müşterinin kaybedilme oranı da azalmaktadır.

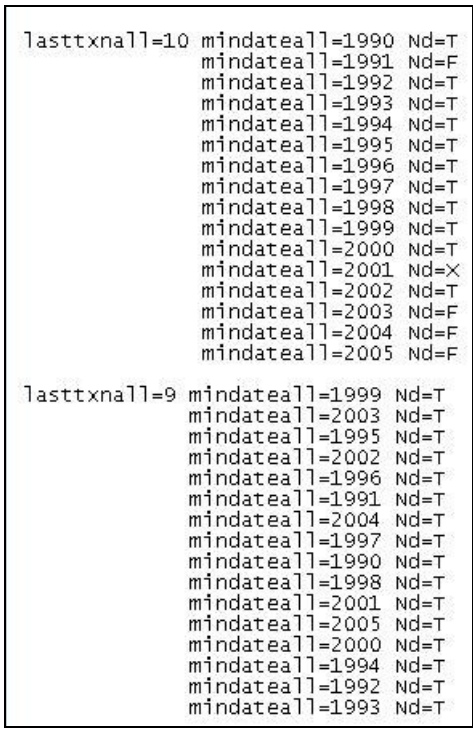


Şekil 4.12 : İkinci analizde eşik değeri 0.25 iken orataya çıkan karar ağacının ilk parçası

Bu da bize, aslında ilk kez 1998 ve öncesi yıllarda hesap açtırmış bir müşteri, 11. ve 12. aylarda kredi kartı ile bir işlem yaptıysa bile, kaybedilme olasılığının yüksek olduğunu göstermektedir.

Analize devam edildiğinde görülecektir ki, en son 10. ve 9. aylarda kredi kartını kullanan müşterinin kaybedilme olasılığı, 1998 öncesinde ilk kez hesap açtırmış olmasından ziyade, 2001 yılında ilk kez hesap açtırmışsa düşmektedir.

9. ay ve öncesi aylar için ise hesap açtırma tarihi ilk kez ne zaman olursa olsun, müşteri yaklaşık olarak 3 ve daha fazla aydır kartını kullanmıyor olduğundan, kaybedilme olasılığı giderek yükselmektedir. Bu da aslında banka açısından beklenen bir sonuçtur. İkinci analizin kurallarını oluşturup test sonuçlarını incelemek için 0.19 eşik değeri seçilmiştir. Bunun sebebi, ağacın çok fazla dallanmasını engellemek ve aşırı detaylardan kaçınmak, aynı zamanda da ağacın çok fazla basit olmasına izin vermemektir.



Şekil 4.13 : İkinci analizde eşik değeri 0.25 iken oratoya çıkan karar ağacının ikinci parçası

Bu analiz sonucunda, kullanılan nitelik sayısı 4 olarak belirlenmiştir. (LAST_TXN_DATE_OF_ALL_ACCOUNTS,
MIN_FIRST_OPEN_DATE_OF_ALL_ACCOUNTS,
LAST_03_MONTH_SUM_ALISVERIS, LAST_03_MONTH_SUM_ALL) Sonuçta ise ortaya Nd=X sayıları hesaba katılmadan 504 adet kural çıkmıştır. Bu kurallar, geri kalan 20,000 kayıtlık veri kümesi üzerinde denenmiştir. Ancak bu kez, ilk analizden farklı olarak bu 504 kuralın hangilerinin daha verimli ya da doğru sonuçlar verdiğini ortaya çıkarmak için her bir kuralın ortaya çıkarttığı doğru ve yanlış sayıları hesaplanmıştır.

Yukarıda bahsedilen, kuralların ne kadar doğru olduğu bilgisinin elde edilmesi işlemi öncelikle öğrenilen veri kümesi üzerinde uygulanmıştır. Bu işlem için, kural numarası, kurala uyan kayıt sayısı ve doğru hedef değerine giden kayıt sayısı bir tabloda birleştirilmiştir. Kurala uyan kayıt sayısı ve hedef değerin doğru olduğu kayıt sayısı arasındaki fark 0 ise, bu kuralın çok iyi sonuç verdiği sonucuna ulaşabiliriz. Bu iki sayı arasındaki fark fazla ise, bunun 2 sebebi olabilir : birincisi Nd=X çıkan kuralların test algoritmasına eklenmemiş olması, ikincisi ise algoritmanın, tüm örnek kümesindeki kayıtların aynı hedef değerlerine gitmediği durumlarda eğer nitelik

listesinde daha fazla seçilebilecek nitelik kalmadıysa, kümedeki en fazla hedef değerini yaprak değer olarak seçmesidir.

Fark değer 0 iken, kurala uyan ve doğru hedefe giden kayıt sayısının maksimumda olduğu 5 adet kural aşağıdaki tabloda görülebilir :

Tablo 4.8 : İkinci analizde kurala uyan kayıt sayısı ve doğru hedefe giden kayıt sayısının birbirine eşit ve maksimum olduğu 5 adet kural

Kural No	Kurala Uyan Kayıt Sayısı	Dogru Hedefe Giden Kayıt Sayısı	Fark
261	281	281	0
284	47	47	0
306	51	51	0
354	110	110	0
455	80	80	0

Kural açıklamaları ise Tablo 4.9'dan görülebilir.
Farkın 0 olup olmadığı göz önüne alınmadığında, kurala uyan kayıt sayısının maksimum olduğu 3 adet kural Tablo 4.10'dan görülebilir (Bu kurallar aynı zamanda en fazla doğru tahminlenen hedef sayısının da maksimum olduğu kurallardır.)

Bu kurallara ilişkin açıklamalar Tablo 4.11'de bulunabilir.Bunlar, sınıflandırmada en önemli olan kurallardır.

En fazla yanlış çıkartma olasılığı olan kuralları bulmak için ise, fark oranı, yani, kurala uyan kayıt sayısı ile doğru hedef değeri tahminlenen kayıt sayısı farkının yine kurala uyan kayıt sayısına oranından faydalanılmıştır. Buna göre, fark oranı en fazla olan kurallar Tablo 4.12'de listelenmiştir. Bu kuralların açıklamaları Tablo 4.13'de görülebilir.

Tüm bu kuralları ve doğruluk oranlarını gözönüne alarak, 20,000 kayıtlık test verisi üzerinde aynı işlemleri uyguladığımızda varılan sonuçlar Tablo 4.14'de görülebilir.

Tablo 4.14'de, fark değerin 0 olup olmadığı gözetilmeden kurala uyan kayıt sayılarının maksimum olduğu halde fark oranı en küçük olan 3 değer seçilmiştir. Görüldüğü üzere, bu kural numaraları öğrenme veri kümesinde ortaya çıkan 3 kural numarası ile aynıdır.

Tablo 4.9 : İkinci analizde kurala uyan kayıt sayısı ve doğru hedefe giden kayıt sayısının birbirine eşit ve maksimum olduğu 5 kuralın açıklama ve hedef değerleri

Kural	Kural Açıklama	Hedef
261	Lasttxnall=8 ve Mindateall=1996	T
284	Lasttxnall=9 ve Mindateall=1995	T
306	Lasttxnall=9 ve Mindateall=1997 ve Mindateopen=1997 ve Last3sumall<>5000-15000	T
354	Lasttxnall=11 ve Mindateall=1996	T
455	Lasttxnall=10 ve Mindateall=1996	T

Tablo 4.10 : İkinci analizde kurala uyan kayıt sayısı ve doğru hedefe giden kayıt sayısının birbirinden farklı ve maksimum olduğu 3 kural

Kural No	Kurala Uyan Kayıt Sayısı	Dogru Hedefe Giden Kayıt Sayısı	Fark
91	782	714	68
186	614	595	19
435	1040	1016	24

Tablo 4.11 : İkinci analizde kurala uyan kayıt sayısı ve doğru hedefe giden kayıt sayısının birbirinden farklı ve maksimum olduğu 3 kuralın açıklama ve hedef değeri

Kural	Kural Açıklama	Hedef
91	Lasttxnall=12 ve Mindateall=2004 ve Mindateopen=2004 ve Last3sumall<>2700-5000	F
186	Lasttxnall=12 ve Mindateall=2005 ve Mindateopen=2005 ve Last3sumall<>1400-2000	F
435	Lasttxnall=7	T

Tablo 4.12 : İkinci analizde kurala uyan kayıt sayısı ve doğru hedefe giden kayıt sayısının oranının maksimum olduğu 3 kural

Kural	Kurala Uyan Kayıt Sayısı	Dogru Hedefe Giden Kayıt Sayısı	Fark	Fark Oran
103	30	16	14	0.466666667
108	31	16	15	0.483870968
403	11	6	5	0.454545455

Fark oranının ve kurala uyan kayıt sayısının maksimum olduğu 3 kural numarası ise Tablo 4.15'de görülebilir.

Tablo 4.13 : İkinci analizde kurala uyan kayıt sayısı ve doğru hedefe giden kayıt sayısının oranının maksimum olduğu 3 kural açıklama ve hedef değerleri

Kural	Kural Açıklama	Hedef
103	Lasttxnall=12 ve Mindateall=1997 ve Mindateopen=1997 ve Last3sumall=900-1400 ve Last3alver<>500-900	F
108	Lasttxnall=12 ve Mindateall=1997 ve Mindateopen=1997 ve Last3sumall=2700-5000 ve Last3alver=2700-5000	F
403	Lasttxnall=11 ve Mindateall=2001 ve Mindateopen=2001 ve Last3sumall=0-200	F

Tablo 4.14 : İkinci analizde kurala uyan kayıt sayısı ve doğru hedefe giden kayıt sayısının oranının maksimum olduğu ve fark oranının minimum olduğu 3 kural

Kural No	Kurala Uyan Kayıt Sayısı	Dogru Hedefe Giden Kayıt Sayısı	Fark	Fark Oran
91	1296	1218	78	0.060185185
186	901	872	29	0.032186459
435	1127	1062	65	0.057675244

Tablo 4.15 : İkinci analizde kurala uyan kayıt sayısı ve doğru hedefe giden kayıt sayısının oranının maksimum olduğu ve fark oranının maksimum olduğu 3 kural

Kural No	Kurala Uyan Kayıt Sayısı	Dogru Hedefe Giden Kayıt Sayısı	Fark	Fark Oran
138	16	0	16	1
218	12	0	12	1
247	13	0	13	1

Bu kurallar, ağaçta, test verisi üzerinde en fazla hataya sebep olan 3 kural olarak tanımlanabilir. Aynı zamanda fark oranının 1 olduğu kurallar da bize test verisi üzerinde denendiğinde hiç bir doğru tahminleme yapılamamış kuralların tablosunu vermektedir. Test verisi üzerinde fark oranı 1 olan tam 70 adet kural bulunmuştur. Yukarıda belirtilen kurallar ve hedef değerleri Tablo 4.16'da görülebilir.

Analizde test ve öğrenme kümesinin doğru sonuçları için yapılan başka bir test ise, bu iki kümedeki doğru hedefe gitme oranlarının karşılaştırılması olmuştur. Her iki kümede, doğru hedefe giden kayıt sayısı ile kurala uyan kayıt sayısı oranlanmış, bu değere "doğru hedefe gitme oranı" adı verilmiştir.

Tablo 4.16 : İkinci analizde kurala uyan kayıt sayısı ve doğru hedefe giden kayıt sayısının oranının maksimum olduğu ve fark oranının maksimum olduğu 3 kural açıklaması ve hedef değerleri

Kural	Kural Açıklama	Hedef
138	Lasttxnall=12 ve Mindateall=1998 ve Mindateopen=2004 ve Last3sumall<>1400-2000	T
218	Lasttxnall=12 ve Mindateall=1989 ve Mindateopen=1989 ve Last3sumall=500-900	T
247	Lasttxnall=12 ve Mindateall=1992 ve Mindateopen=2000	T

Daha sonra bu değerler iki küme için yanyana konarak aralarındaki fark değerine bakılmıştır. Tablo 4.17'de bu karşılaştırmanın sonuçları bulunmaktadır

Tablo 4.17 : İkinci analizde doğru hedefe gitme oranı farkı ve kural sayıları

Doğru hedefe gitme oranı farkı	Kural Sayısı	Açıklama
0	70	Hem öğrenme hem test kümesinde bu oranlar yaklaşık 1'dir.
1	62	Öğrenme kümesindeki oran 1 iken test kümesindeki oran 0'dır.
0-1 arası	214	Öğrenme kümesindeki oran, test kümesindeki orandan büyüktür.
0'dan küçük	88	Test kümesindeki oran, öğrenme kümesindeki orandan büyüktür.

Bu tablodan çıkarılabilecek sonuçlar şunlardır : 70 adet kuralın hem öğrenme hem de test kümesinde doğru hedefe gitme oranı eşit ve %100'e yakındır. Bu kurallar, hangi veri kümesinin üzerinde test yapıldığından bağımsız olarak doğru sonuçlar verdiğinden, sınıflandırmanın en iyi yapıldığı kurallardır.

Benzer şekilde, 62 adet kural, sadece öğrenme kümesi için geçerli olmuştur, test kümesinde tümü geçersiz çıkmıştır. 214 kuralda, öğrenme kümesinde elde edilen doğruluk başarısı, test kümesinde elde edilen başarıdan daha fazladır, ki bu normalde karar ağaçlarının farklı veri kümeleri üzerinde test edilmesinde beklenen bir durumdur. Çünkü, her bir veri kümesi kayıdı, farklı bir kural çıkarma olasılığı demektir. Bu veri kümeleri birbirinden çok farklı sonuçlar çıkartırlarsa karar ağacı aslında daha büyük bir örnek kümesi ile oluşturulmalıdır anlamına gelmektedir. Bu tablodaki son satır ise, ilginç bir istatistiği daha ortaya koymaktadır. 88 adet kuralda, öğrenme kümesinde elde ettiğimiz kuralın aynı kümede test edildiğinde doğru

sonuçları veren kayıt sayısı, test kümesinde test edildiğinde doğru sonuçları veren kayıt sayısından düşük çıkmıştır. Bu da, karar ağacından beklenen sonuçların test edilirken, düşünüldüğünden farklı kurallara ve sonuçlara bizi ulaştırabileceği anlamına gelmektedir.

Son yapılan bu testlerin amacı, gereksiz ya da yanlış sonuçlar veren kuralların numaralarını bulup sonradan-budama yöntemiyle bu kuralların üzerinde işlem yapılabilmesidir. Buna göre, kurallara doğruluk yüzdeleri verilerek, ya kuralın tamamen iptal edilmesi, ya da hedef değerinin değiştirilerek yeniden analiz yapılması gerekmektedir.

Toplamdaki doğru bulunan kayıt sayılarını karşılaştırdığımızda Tablo 4.18'deki sonuçlara ulaşırız :

Tablo 4.18 : İkinci analizde toplam kurala uygun olma ve doğru hedefe gitme sayı ve oranları

	Kurala uyan kayıt	**Doğru hedefe giden kayıt**	**Kurala uygunluk yüzdesi**	**Doğru hedefe gitme yüzdesi**
Öğrenme Kümesi (10,000)	9879	8871	0.9879	0.8871
Test Kümesi (20,000)	19317	12928	0.9659	0.6464

Bu sonuçlar, bize aslında karar ağacında bulunan kuralların, test verisi üzerinde denendiğinde de %96 oranında aynen bulunduğunu göstermektedir. 20,000 kayıtlık veride yalnızca %4'lük bir oran kadarının kural tablosuna uymadığı göslenmiştir. Doğru hedefe gitmenin tahminlenmesi ise, %65 civarında çıkmıştır. Bu da bize, en fazla güvenilir kurallara bakarak veri analizini yapıp gereken sonuçlara ulaşabileceğimiz kadar iyi bir oran yüzdesini vermektedir.

5. SONUÇLAR ve ÖNERİLER

Yapılan karar ağacı uygulamasında denenen kurallardan sonra ortaya şu sonuçlar çıkarılmıştır :

- Karar ağaçlarında eşik değeri kullanılırsa, eşik değeri arttıkça, kullanılan niteliklerin sayısı azalır. Ağaç daha basit ve detaysız olur. Ancak, belirsiz hedef değerlerine götüren kural sayısı genel bir artış gösterir. Böylece, eşik değerinin seçimi, analizin ne amaçla yapıldığına bağlıdır.
- Karar ağaçları iyi bir sınıflandırma yaptılarsa, hangi veri kümesi üzerinde uygulandığına bağlı olmadan yaklaşık eşit yüzdelerle doğru sonuçlar verirler.
- Test veya öğrenme verisinin boyutlarının büyütülmesi, verinin nasıl bir özellik gösterdiğine bağlı olarak çıkan sınıflnadırmanın ne kadar verimli olduğunu etkiler. Bu tezde yapılan denemelerden sonra, bu boyutların büyütülmesi, doğruluk oranında aşırı bir değişiklik yapmamış olmasına rağmen, veri hakkında daha detaylı bilgiler elde edilmesini sağlamıştır.
- İlk analiz aşamasında, doğruluk oranının maksimum olduğu 0.37 eşik değeri için çıkarılan kural tablosu incelendiğinde aşağıdaki sonuçlara ulaşılmıştır :
    - Verinin alındığı tarih 12. ay olduğundan, son hesap hareketi tarihi 12'ye yakın olan aylarda müşterinin kaybedilme olasılığı az, diğerlerinde yüksektir.
    - Son hesap hareketi tarihi 12. ayken müşteri son 3 ayda hiç alışveriş yapmamışsa genel olarak kaybedilme olasılığı oldukça yüksektir.
    - Son hesap hareket tarihi 11.ayken, müşteri son 3 ayda göreceli olarak az harcama yaptıysa veya çok fazla harcama yaptıysa kaybedilmiştir. Çok harcama yapan müşterinin kaybedilmesinin sebebi olası bir dolandırıcılık konusu olarak düşünülebilir.
- İkinci analiz aşamasında, 0.25 eşik değeri için çıkarılan kural tablosu incelendiğinde aşağıdaki sonuçlara ulaşılmıştır :

- Son hesap hareketi tarihi 11 ve 12. aylarda olan bir kişinin kaybedilme olasılığı, eğer ilk hesap açtırma tarihi 1998 yılından önce ise, daha fazladır. Bu durumda 1998 tarihi öncesinde ilk kez bankanın kredi kartı müşterisi olan bir kişinin, kredi kartını kullanıyor olsa bile, banka için fazla sadık bir müşteri olmadığı düşünülebilir.
- Son hesap hareket tarihi 10.ay olan bir müşteri, ilk kez hesabını 2005 yılı ve sonrasında açtırdıysa, bu müşterinin kaybedilme olasılığı düşüktür. 2005-2004 yılı öncesinde müşteri haline gelmiş biri için, 10. ayda yapılan son hareket, kaybedilme olasılığını azaltmamaktadır.
- 10.ay ve öncesinde son hareket tarihi olan bir müşterinin kaybedilme olasılığı, diğer nitelikleri ne olursa olsun daha yüksektir.

- İkinci analiz aşamasında, 0.19 eşik değeri için çıkarılan kural tablosu incelendiğinde aşağıdaki sonuçlara ulaşılmıştır :
  - Karar ağacı, müşterinin tüm hesaplarının içindeki minimum açılış yılı ile, açık hesaplarının içindeki minimum açılış yılı, birbirine eşit olduğu zamanlarda daha detaylı bir yapı ortaya çıkarmaktadır, yani, hedef değerini elde etmek için diğer niteliklerden de faydalanmak zorunda kalmaktadır.
  - Genel olarak, tüm diğer nitelikler için, son 3 aylık toplam harcama miktarı göreceli olarak daha düşük olduğu zamanlarda, müşterinin kaybedilme olasılığı yükselmektedir.
  - Son 3 aylık toplam harcama ve alışveriş miktarları göreceli olarak çok yüksek olduğu zamanlarda ise müşterinin kaybedilme olasılığı yine yükseliş göstermektedir.
  - Tüm analizlerde görülmüştür ki, son hesap hareketi tarihi 6. ve 7. ay olan müşteriler, genel olarak diğer nitelikleri ne olursa olsun kaybedilmektedirler.
  - 8 ve 9. aylarda son kez işlem yapmış olanların da kaybedilme olasılığı yüksektir. Bu da bize, mevcut zamandan en az 3 ay önce son kez işlem yapmış kişiler için bankanın önlem alması gerektiğini göstermektedir.

Bu çalışma, aşağıda yapılabilecek olan işlemlerle genişletilebilir :

- Nd=X olan belirsizlik gösteren kuralları kural tablolarına katabilmek için, kesin hedef değerleri yerine olasılıklı hedef değerleri kullanılabilir. Böylece test işlemlerinde de tüm kayıtlar için olasılıkların çarpımlarıyla oluşan bir olası hedef değeri denenebilir. Belli olasılıkların üzerinde kaybedilme riski olan müşteriler için ise önlem alınmaya başlanabilir.
- Algoritma her türlü veri kümesi için çalışabilecek şekilde parametrik hale getirilerek, çıkan kuralları test etmek için ayrıca manuel olarak kural girilen programlar yerine, bu kuralların otomatik olarak, analiz sonrasında test edilmesi sağlanabilir.
- Karar ağaçları yerine farklı bir veri madenciliği yöntemi kullanılabilir ve ortaya çıkarılan sonuçlar karşılaştırılıp hangi modelin böyle bir veri kümesi için daha uygun olup olmayacağı tartışılabilir.
- Verinin analizi öncesinde kullanılan histogramlar ve yerine koymalı eksik tamamlama yöntemleri yerine farklı yöntemler kullanılarak aynı karar ağacı yapısında sınanıp sonuçlar karşılaştırılabilir.
- Tezin son aşamasında başlatılan sonradan-budama yöntemi ile kurallar budanıp, sonuç karşılaştırması yapılabilir.

KAYNAKLAR

[1] **Han, J. and Kamber , M.,** 2001. Data Mining Concepts and Techniques, Academic Press. New York.

[2] **Yan, L. and Miller, J**. **and Mozer, M**. **and Wolniewicz, R**., İmproving Prediction of Customer Behaviour in Nonstationary Environments.

[3] **Richeldi, M. and Perrucci , A.,** 2002. Churn Analysis Case Study, Telecom Italia Lab. Torino. 3-6

[4] **Moore, A.W**., 2001. Decision Trees, Carnegie Mellon University, USA.

[5] **Karamustafa, K**. **ve Biçkes, D**., Kredi Kartı Sahip ve Kullanıcılarının Kredi Kartı Kullanımlarını Değerlendirmeye Yönelik Bir Araştırma: Nevşehir Örneği, Erciyes Üniversitesi, Nevşehir, 92-95

[6] **Rud, O.P. ,** 2001. Data Mining Cookbook: Modeling Data for Marketing, Risk and Customer Relationship Management, John Wiley & Sons, Inc. Canada.

EK-A

Uygulamanın kaynak kodu ve veri kümeleri ile analiz tablolarının tümü CD-ROM'da bulunmaktadır.

ÖZGEÇMİŞ

Tuğba Tosun 19/08/1980 tarihinde İzmir'de doğdu. 2002 yılında İzmir Yüksek Teknoloji Enstitüsü Bilgisayar Mühendisliği'nden mezun oldu. 2003 yılında İstanbul Teknik Üniversitesi Bilgisayar Mühendisliği bölümünde yüksek lisans yapmaya başladı. 2004 yılından beri Yapı Kredi Bankası'nda yazılım uzmanı olarak çalışmaktadır.