YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

FİNANSAL VERİ MADENCİLİĞİ

Fizikçi Ünal ARAS

FBE Matematik Müh. Anabilim Dalında
Hazırlanan

YÜKSEK LİSANS TEZİ

Tez Danışmanı: Yrd. Doç. Dr. Nilgün Güler BAYAZIT

İSTANBUL, 2008

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ

$y(p)$	Elde edilen çıkış değeri
$y_d(p)$	Beklenen çıkış değeri
p	İterasyon adımı
$x(p)$	Giriş değerleri
w	Girişlerin ağırlıkları
θ	Eşik değeri
η	Öğrenme katsayısı

KISALTMA LİSTESİ

ARFF	Attribute Related File Format
ÇKA	Çok Katmanlı Algılayıcı
LR	Lojistik Regresyon
MİY	Müşteri İlişkileri Yönetimi
SQL	Structured Query Language
Weka	Waikato Environment for Knowledge Analysis

ŞEKİL LİSTESİ

ÇİZELGE LİSTESİ

ÖNSÖZ

Günümüzde veritabanı sistemlerinin kullanımının artması, veritabanlarının hacimlerindeki büyük artışı da beraberinde getirmiştir. Elde edilen büyük veri yığınlarının analizinin gerçekleştirilmesinde, mevcut veri analizi yöntemleri yeterli gelmemeye başlamıştır. Bu nedenle yapılan arayışlar neticesinde, veri tabanlarından bilgi çıkarımı süreci olan veri madenciliği kavramı ortaya çıkarılmıştır.

İşletmelerin birbirine benzeyen ürünler ve hizmetler sunduğu günümüzde, müşteri odaklı çözümlerin üretilmesi adeta bir zorunluluk halini almıştır. Bu çözümlerin başında gelen Müşteri İlişkileri Yönetimi (CRM), aynı zamanda veri madenciliği tekniklerinin yoğun olarak kullanıldığı alanlardan biridir. Finans alanında önemli bir yere sahip olan bankacılık sektörü, dinamik yapısı ve sektör içi rekabetin yoğun yaşanması gibi etkenlerle Müşteri İlişkileri Yönetimi'nin önemli uygulama alanlarından birisi olmuştur.

Bu doğrultuda tezin birinci bölümünde veri madenciliği kavramı tanımlanmış; ikinci bölümde veri madenciliğinin önemli uygulama alanlarından olan Müşteri İlişkileri Yönetimi ile ilgili bilgi verilmiş; üçüncü bölümde veri madenciliğinde öne çıkan modeller üzerinde durulmuş; son bölümde ise çeşitli veri madenciliği teknikleri kullanılarak, bir bankaya ait kredi kartı ve kredilerin statüsü verileri üzerinde müşteri profili ve müşteri segmentasyonu gerçekleştirilmiştir.

Bu çalışmanın tamamlanmasında yardımlarını ve desteğini esirgemeyen saygıdeğer hocam Yrd. Doç. Dr. Nilgün Güler BAYAZIT'a teşekkürü bir borç bilirim.

Ocak 2008 Ünal ARAS

FİNANSAL VERİ MADENCİLİĞİ

ÖZET

Veri madenciliği, büyük veritabanlarında yer alan verilerdeki gizli eğilimlerin ortaya çıkarılması sürecidir. Finans, pazarlama, telekomünikasyon gibi birçok alanda veri madenciliği teknikleri kullanılmaktadır.

Finans alanında önemli bir yere sahip olan bankacılık sektörü, veri madenciliğinin yoğun olarak kullanıldığı sektörlerdendir. Banka müşterilerinin, sahip oldukları kredi kartı tipine ve aldıkları kredinin geri ödenebilirliğine bakılarak müşteri profillerinin ve müşteri segmentasyonlarının belirlenmesi süreci, veri madenciliği teknikleri ile başarılı bir şekilde gerçekleştirilmektedir.

Bu tez çalışmasında, çeşitli veri madenciliği teknikleri kullanılarak, bir bankaya ait veri tabanından elde edilen kredi kartı ve kredilerin statüsü verileri üzerinde müşteri profili ve müşteri segmentasyonu uygulamaları gerçekleştirilmiştir. Sonuç olarak, veri madenciliği sürecinin, müşteri profili ve müşteri segmentasyonu uygulamalarını başarı ile gerçekleştirebildiği görülmüştür.

Anahtar Kelimeler: Veri madenciliği, müşteri segmentasyonu, müşteri profili.

FINANCIAL DATA MINING

ABSTRACT

Data mining is the process of bringing out appropriate information that is stored in huge databases. Data mining techniques are extensively used in many areas, including Finance, Marketing and Telecommunication.

Data mining is widely used by in banking sector which is the core component of Finance Industry. Data mining techniques are being been successfully used to identify bank customers' profiles and segmentations by analyzing their credit card types and their ability to pay back their outstanding credit card or loan debts.

In this thesis, I have used various data mining techniques to carry out costumer profiling and segmentation by using credit card and loan data that belongs to one individual bank's database.

To conclude, the process of data mining has been successfully applied to identifying costumer profiles and segmentations.

Key Words: Data mining, customer segmentation, customer profiling.

1. GİRİŞ

Bilgisayar sistemlerinde gerek işlemcilerin hızlarının gerek de disklerin kapasitelerinin hızla artması, bilgisayarların daha büyük miktarlardaki veriyi daha kısa bir sürede işleyebilmelerine imkan sağlamaktadır. Aynı zamanda bilgisayar ağlarındaki hızlı gelişme ile veriye başka bilgisayarlardan da hızlı bir şekilde erişebilmek mümkündür. Bilgisayarların fiyatlarındaki ucuzlama ile orantılı kullanıcı sayısının da giderek artması, verinin doğrudan sayısal olarak toplanması ve saklanmasını mümkün kılmaktadır.

Verinin hızlı bir şekilde artması, güçlü veri analizi çözümlemelerini zorunlu kılmaktadır. Aksi halde eldeki verinin çok büyük olmasına karşın ortaya çıkan bilgi çok az veya yetersiz olabilecektir. Yeterli bilgi çıkarımının olmadığı böyle bir süreçte şüphesiz eldeki verinin büyüklüğü de pek bir anlam ifade etmeyecektir.

Bilgi, verinin bir amaca yönelik işlenmesi ile elde edilir. Veriyi bilgiye çevirme olarak tanımlayabileceğimiz veri analizi, işletmeler için son derece önemli bir kavramdır. Geleneksel sorgulama veya raporlama araçlarının, verinin analizi sürecinde yetersiz kalması nedeniyle yeni analiz yöntemleri üzerinde çalışmalar yapılmıştır. Bu araştırmalar sonucunda ortaya çıkan veri madenciliği, veriden bilgi çıkarımı sürecinde kullanılan temel çözümlemelerden biri olmuştur.

İşletmelerin müşterilerini ayrıştırmak, onlarla etkileşimli iletişim kurmak gibi müşteri odaklı çözümler arayışında olması, Müşteri İlişkiler Yönetimi yaklaşımının ortaya çıkmasına zemin hazırlamıştır. Özellikler günümüzün rekabetçi piyasa ortamında, farklı ürün ve hizmetler ile dikkat çekmek isteyen birçok işletme, Müşteri İlişkileri Yönetimi yaklaşımını etkin bir şekilde kullanmaktadır. Müşteri İlişkileri Yönetimi'nin yoğun olarak kullanıldığı alanların başında finans sektörü gelmektedir.

Müşteri İlişkileri Yönetimi'nin temel bileşenlerinden olan müşteri profili ve müşteri segmentasyonu, veri madenciliğinin yaygın olarak kullanıldığı uygulama alanlarındandır.

Bu tez, özellikle son dönemlerde finans sektörü tarafından veri analizi sürecinde etkin bir şekilde kullanılan veri madenciliği kavramı hakkında bilgi vermeyi amaçlamaktadır. Bu amaçla veri madenciliği yaklaşımı tanıtılmış, yoğun olarak kullanıldığı Müşteri İlişkileri Yönetimi kavramı ile ilgili bilgi verilmiş ve bir bankanın veritabanı kullanılarak veri madenciliği tekniklerinin uygulanması hedeflenmiştir.

Tezin birinci bölümünde veri madenciliği kavramı açıklanmış, ikinci bölümde Müşteri

İlişkileri Yönetimi hakkında bilgi verilmiş, üçüncü bölümde veri madenciliğinde öne çıkan modeller açıklanmış ve son bölümde ise bir bankanın kredi kartı ve kredi statüsü verileri üzerinde veri madenciliği teknikleri uygulanmıştır.

2. VERİ MADENCİLİĞİ (DATA MINING) KAVRAMI

2.1 Veri Madenciliği Kavramına Genel Bakış

Veritabanı teknolojisinde sağlanan hızlı gelişme, veritabanı, veri ambarı ve diğer bilgi depolarında toplanan verinin hacmindeki büyük artışı da beraberinde getirmiştir. Eldeki verinin büyüklüğü ve karmaşıklığının artması sonucunda, büyük miktardaki veri içerisinde yer alan gizli örüntülerin bulunmasında geleneksel çözümleme yöntemleri yetersiz kalmış, verinin daha iyi çözümlenme yöntemleri ile işlenmesi gerekliliği artmıştır. Bu amaçla kullanılan yöntemlerin başında ise veri madenciliği uygulamaları gelmektedir.

Veri madenciliği; bir işletmede tüm verilerin toplandığı yazılım ve donanımı ifade eden veri ambarlarında yer alan verilerdeki gizli eğilimleri, bu eğilimlerin birbirleriyle ilişkilerini, ortaya çıkan ilişkilerin nedenlerini ve verilerin nasıl bir seyir gösterdiğini ortaya çıkaran yöntemler topluluğu, modellemeler ve teknikler olarak görülebilir (Özmen, 2003).

Kavram, işletmelerin veritabanlarındaki saklı bilgilerin bulunup ortaya çıkarılmasını içerdiğinden, altın ya da kömür madenciliğine benzetilmiştir (Rygielski, 2002).

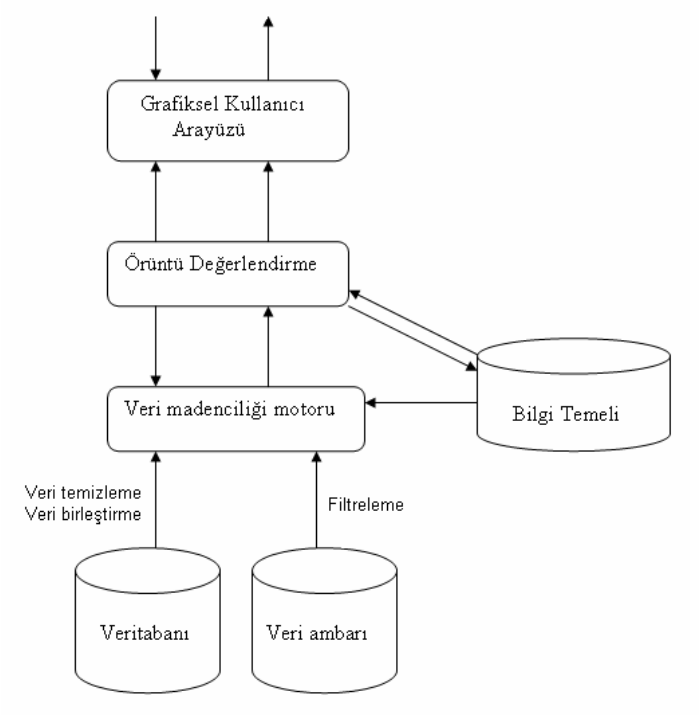
Veri madenciliği geçmiş verileri, gelecek ile ilgili etkin kararlar verebilmek için bir rehber olarak kullanmaktadır. Bu ise geçmiş verilerin yüksek açıklama gücüne sahip modeller ile tanımlanması ve bu modellerin muhtemel gelecek veri çıkışlarını tahmin etmesi ile mümkün olmaktadır.

Veri madenciliği mimarisi tipik olarak aşağıda belirtilen bileşenlerden oluşmaktadır (Han ve Kamber, 2001):

- Veritabanı, Veri Ambarı veya Diğer Bilgi Depoları: Bu aşamada veri; veritabanı, veri ambarı veya diğer bilgi depolarında toplanmaktadır. Veri üzerinde veri temizleme ve veri bütünleştirme (data integration) işlemleri gerçekleştirilebilir.
- Veritabanı veya Veri Ambarı Sunucusu: Veritabanı veya veri ambarı sunucusu, kullanıcının yapacağı veritabanı işlemine göre, gerekli olan ilişkili verinin ortaya çıkartılmasından sorumludur.
- Bilgi Temeli: Bu kısım, araştırma için bir rehber ya da ilginç örüntülerin değerlendirildiği aşama için bir temel oluşturmaktadır. Niteleyiciler (attributes) ya da niteleyicilerin değerleri, farklı düzeylerde organize edilmek için kullanılır.
- Veri Madenciliği Motoru: Veri madenciliği motoru, veri madenciliği sistemlerinde önemli bir yere sahip olan ilişki analizi, sınıflandırma ve kümeleme analizi için son

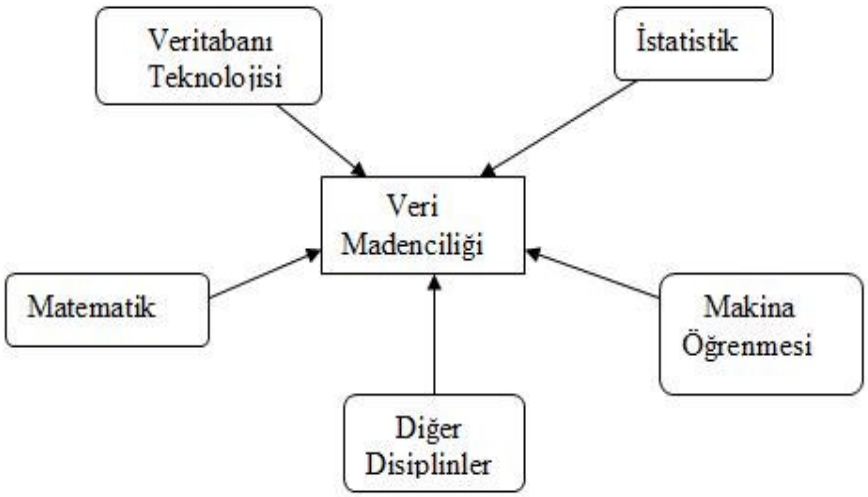
derece gerekli bir kısımdır.

- Örüntü Değerlendirme: Bu bileşen tipik olarak, ilgi çekici olan örüntülerin ortaya çıkartıldığı aşamadır. Bu aşama uygulanan veri madenciliği metoduna bağlı olarak veri madenciliği modülü içerisinde de yer alabilir.
- Grafiksel Kullanıcı Arayüzü: Bu aşama, kullanıcı ile veri madenciliği sistemi arasındaki bağlantının sağlandığı aşamadır. Kullanıcının veritabanı ile etkileşimine izin verilir. Bu etkileşim, kullanıcı tarafından veri madenciliği araştırmasına katkı sağlanması amacıyla sisteme bilgi girişi yapılması şeklinde gerçekleştirilebilir. Ayrıca bu bileşen kullanıcıya, veritabanı ile veri ambarı şemalarını taramayı ve değişik formlardaki örüntülerin görüntülenmesini sağlamaktadır.



Şekil 2.1 Veri madenciliği mimarisi (Han ve Kamber, 2001).

Veri madenciliği bilgi çıkarımı sürecini matematik, istatistik, makine öğrenmesi, veritabanı teknolojisi vb. birçok disiplinden faydalanarak gerçekleştirmektedir (Şekil 2.2).



Şekil 2.2 Veri madenciliğinin etkileşim içerisinde bulunduğu disiplinler.

2.2 Veri madenciliğinin Finans Sektöründeki Yeri ve Önemi

Veri madenciliğinin, büyük veri yığınlarından anlamlı örüntüleri çıkarabilmesindeki başarısı, finans, tıp, telekomünikasyon gibi birçok farklı alanda veri madenciliği tekniklerinin kullanılmasını da beraberinde getirmiştir.

Finans alanında önemli bir yere sahip olan bankacılık sektörü, veri madenciliği sürecinin yoğun olarak kullanıldığı sektörlerden biridir. Veri madenciliği tekniklerinin bankacılık sektöründe kullanıldığı alanlardan bazıları şunlardır:

- Kredi Pazarlama: Veri madenciliği yöntemleri ile müşterilerin profilleri saptanarak müşteri segmentleri belirlenebilmektedir. Buna göre veri madenciliği ile segmente yönelik pazarlama faaliyetleri gerçekleştirilerek karlılığı arttırmak mümkün olmaktadır (Shaw, 2001).
- Kredi Riskinin Değerlendirilmesi: Veri madenciliğini benzer işleve sahip yöntemlerden ayıran en önemli farkı, geçmiş kredi alan müşterilerin yaptığı işlemleri baz alarak tahmine dayalı bir model ortaya koyabilmesidir. Bu sayede müşteriler tarafından ödemenin geciktirilip geciktirilmeyeceği ile ilgili tahmin önceden yapılabilir (Stefanowski ve Wilk, 1997; Özmen, 2003).
- Ürünlerin Yaşam Eğrisinin Analizi: Müşteri davranışlarına göre sunulan ürünlerin yaşam eğrisinde hangi aşamada olduğu veri madenciliği ile belirlenebilir. Böylece

müşteri davranışları temelinde pazarlama faaliyetinin yürütülmesi mümkün olmaktadır (Rygielski, 2002).

- Kredi Kartı Harcamalarına Göre Müşteri Gruplarının Belirlenmesi: Müşteriler, kredi kartı harcamalarına göre gruplandırılarak, her müşterinin harcama alışkanlıkları belirlenir. Böylece banka tarafından, müşterinin kredi kartını daha fazla kullanmasına yönelik promosyonlar gerçekleştirilebilir.

Veri madenciliğinin finans ve bankacılık alanlarındaki kullanımı Çizelge 2.1'de özetlenmiştir.

Çizelge 2.1 Veri madenciliği tekniklerinin finans/bankacılık sektöründe kullanıldığı alanlar.

Sektör	**Uygulama Alanı**
Bankacılık/Finans	• Kredi kartı dolandırıcılıklarının tesbiti • Kredi kartı harcamalarına göre müşteri gruplarının belirlenmesi • Kredi taleplerinin değerlendirilmesi • Risk analizi ve risk yönetimi • Müşteri karlılığının analizi • Hisse senedi fiyat tahmini • Dış pazarlardaki fiyat değişimi tahmini • Kur/Faiz oranı tahmini

3. MÜŞTERİ İLİŞKİLERİ YÖNETİMİ

3.1 Müşteri İlişkileri Yönetimi Kavramı

Günümüzde pazar global bir özellik gösterirken ürünler ve servisler birbirine çok benzer olup çok sayıda da tedarikçileri bulunmaktadır. Pazarın büyük ve karmaşık olması nedeniyle toplu pazarlama maliyeti fazla olmakta ve bu nedenle yatırımların geri dönüşleri sık sık sorgulanmaktadır. Bu nedenle uzun vadede işletmelerin, müşterileri seçme ve yönetmeye yönelik müşteri odaklı bir hizmet anlayışına olan gereksinimleri artmıştır. Müşteri İlişkileri Yönetimi (MİY) bu gerekliliğin bir sonucu olarak ortaya çıkmış bir pazarlama stratejisidir.

Temelde MİY, şirketlerin mevcut ya da potansiyel müşterilerinin gereksinimlerini tahmin ederek bu gereksinimlere cevap verebilecek ayıklanmış temiz bilgiyi kullanan bir yaklaşımdır. Pazarlamacılar müşterileri ile ilgili her türlü bilgiyi MİY sayesinde elde etmekte ve böylece bilgi teknolojileri ile çok sayıda parçaya ayrılan veri, birçok pazarlama imkanı sağlayabilmektedir (Child, 2003).

MİY'nin işlerlik kazanması ve başarılı olabilmesinde müşterileri tanımak, ayrıştırmak, onlarla etkileşimli iletişim kurmak, özelleştirilmiş ürün ve hizmet sunmak son derece önemli bir yer tutmaktadır (Özmen, 2003). Bu hedeflere ulaşmada kullanılan yöntemler arasında en çok tercih edileni veri madenciliği uygulamalarıdır.

Müşteriler ile ilgili veri yığınlarının çok büyük miktarlara ulaşması ve bu verilerin anlamlı bilgilere dönüştürülme gerekliliği, veri madenciliğini müşteri ilişkileri stratejilerinin geliştirilmesinde önemli bir kavram olarak ortaya çıkarmaktadır. Müşteriler ile ilgili bilgilerin analizi ve gizli örüntülerin veri madenciliği uygulamaları ile keşfedilmesi, geçmiş müşteri tüketim alışkanlıklarının anlaşılmasını ve böylece örüntülerin tanımlanarak stratejik kararların alınabilmesini sağlamaktadır (Rygielski, 2002).

MİY, müşteri davranışlarını doğru anlayabilmeyi gerektirmektedir. Müşteri davranışlarını anlayabilmek ise işletme ile müşteri arasındaki ilişkinin aşamalarını ifade eden müşterilerin yaşam eğrileri ile bağlantılıdır. Söz konusu yaşam eğrisinin dört aşaması bulunmaktadır (Freeman, 1999):

i) Potansiyel Müşteriler: Henüz müşteri olmamış ancak hedef kitle içerisinde yer alan müşteriler.

ii) Tepki Verenler: Olası müşteriler arasında işletmenin ürün ya da hizmetine ilgi göstermiş kişiler.

iii) Aktif Müşteriler: İşletmenin ürün ya da hizmetini halihazırda kullanan kişiler

iv) Eski Müşteriler: İşletmeyi terk edip rakip ürün ya da hizmeti satın almaya başlamış olanlar; hedef kitle içerisinde yer alma özelliklerini yitirmiş olanlar.

Veri madenciliği ile müşteriler hakkında elde edilen bilgiler artmakta ve pazarlama alanında önemli kararların alınması kolaylaşmaktadır. Bu durum ise MİY'nin başarılı bir şekilde yürütülebilmesini sağlamaktadır.

MİY, müşteri segmentasyonu ve müşteri profilleme süreçlerini kapsamaktadır.

3.2 Müşteri Segmentasyonu

Benzer özelliklerine ya da eğilimlerine göre müşterileri segmentlere yerleştirmeye segmentasyon denilmektedir. Segmentasyon müşterilerle daha etkili bir iletişim kurmak için kullanılan bir yoldur. Segmentasyon işlemi verideki müşteri gruplarının (segment ya da küme olarak adlandırılır) özelliklerini tanımlamaktadır. Müşteri segmentasyonu, belirlenen müşteri gruplarına göre her bir müşteriyi sınıflandırmak için yapılan hazırlık aşamasıdır.

Segmentasyon, günümüzün parçalı tüketici pazarıyla başa çıkabilmek için gereklidir. Pazarlamacılar segmentasyon kullanarak yeni fırsatların keşfedilmesinde ve kaynakların yönlendirilmesinde daha etkili olmaktadırlar. İyi bir segmentasyon yapmanın zorlukları şöyledir (Bounsaythip ve Runsala, 2001):

- Verinin Kalitesi ve Uygunluğu: Verinin kalitesi ve uygunluğu, anlamlı segmentlerin oluşturulması için önemlidir. Eğer şirketin yetersiz ya da gereğinden fazla müşteri verisi varsa, bu durum karmaşık ve zaman kaybı olan analizlere yol açabilir. Benzer olarak eğer veri yeterli derecede iyi düzenlenmemiş ise (farklı formatta, farklı kaynaklardan) ilgi çekici bilgiler çıkartmak zor olacak ve oluşturulan segmentasyon sonuçları etkili bir şekilde uygulanmak için fazla karmaşık olabilecektir. Bilhassa çok fazla segmentasyon değişkeni kullanmak, karar alma sürecinde karışıklığa neden olabilir. Görünüşte etkili olan değişkenler açıklayıcı olamayabilir. Bu problemler, yetersiz müşteri verisinden kaynaklanmaktadır.
- Sezgi: Veri her ne kadar bilgilendirici olsa da pazarlamacıların, doğru veri analizi için sürekli yeni segmentasyon hipotezleri üretmeleri gerekmektedir.
- Daimi Süreç: Yeni müşteriler eklendikçe segmentasyonun güncelleştirilmeye ve geliştirilmeye olan ihtiyacı artmaktadır. Müşterilerin davranışlarına göre oluşturulan verimli segmentasyon stratejileri yine müşterilerin davranışlarından etkilenir. Hatta,

geri dönüşün çok hızlı olduğu e-ticaret ortamında segmentasyon, neredeyse günlük güncelleme gerektirir.

- Aşırı Segmentasyon: Bir segmenti ayrı bir segment olarak değerlendirebilmek çok küçük ve/veya yetersiz olabilir.

3.3 Müşteri Profilleme

Bir şirket, muhtemel tüm müşterileri eşit olarak hedeflemek ya da herkese aynı teşvik edici teklifleri sunmak yerine, müşterilerinin kişisel ihtiyaçlarına ve satın alma özelliklerine göre karlı bir grubu hedefleyebilir. Bireylerin demografik özelliklerine, yaşam tarzlarına, geçmiş davranışlarına göre ileri dönem değerini tahmin eden bir model yapılarak bu başarılabilir. Oluşturulan model en karlı müşteri grubunu korumak ve artırmak için müşteriyi hatırlama ve kaydetmeye odaklanır. Buna müşteri davranışı modelleme ya da müşteri profilleme denilmektedir. Müşteri profilleme pazarlama birimleri için müşterilerinin özelliklerini daha iyi tanımayı ve anlamayı sağlayan bir araçtır.

Müşteri profilleme, pazarlamacıların mevcut müşterilerine daha iyi bir servis sunabilmesi ve bu müşterileri ellerinde tutabilmesi için temel araçtır. Müşteri profilleme, müşterileri yaş, gelir ve yaşam tarzı gibi niteliklerine göre tanımlamaktır. Profilleme demografik ve davranışsal müşteri bilgisinin bir araya getirilmesiyle oluşturulur. Müşteri profilleme, mevcut veriye bağlı olarak yeni müşterileri tahmin etmekte veya mevcut kötü müşterileri belirlemede kullanılabilir. Aynı zamanda müşteri profili çıkartılması ile benzer satın alma özelliklerine sahip müşterilerin gruplara ayrılması gerçekleştirilebilir. Amaca bağlı olarak, projeye uygun olan profil seçilmelidir (Bounsaythip ve Runsala, 2001).

Müşteri profillemenin uzun dönemde getirisi müşteriyi tanımayı müşteriyle otomatik etkileşime dönüştürmesidir. Bu işler için günümüz pazarı birçok alanda süreçlere ve Bilgi Teknolojisi (BT) araçlarına ihtiyaç duymaktadır. Bu araçlar veriyi bir araya getirmede, pazarla ilgili bilgiye ulaşma işlemini basitleştirmede ve pazarlama kampanyalarını planlamakta kullanılmaktadır (Bounsaythip ve Runsala, 2001).

3.4 Müşteri İlişkileri Yönetiminin Faydaları

MİY'nin işletmeler için önemli kazanımları vardır. Bu kazanımları Brown (2000) tarafından şöyle özetlenmiştir:

- Müşteri gereksinimlerine odaklanmayı sağlayarak, getirisi fazla olan müşterileri hedef

almayı sağlamak,

- Müşterilerin davranışlarının doğru bir şekilde analiz edilmesiyle hedef kitleye sunulacak reklam maliyetlerinin düşürülmesini sağlamak,
- Bir ürünü geliştirmek için harcanan pazarlama sürecini kısaltmak,
- Yapılmış bir kampanyanın verimliliğini ölçmeyi kolaylaştırmak.

4. TEMEL VERİ MADENCİLİĞİ TEKNİKLERİ

Veri madenciliği büyük veritabanları arasındaki ilişkileri ve örüntüleri, kullanıcıların sorgularını kullanarak kullanışlı bilginin elde edilmesi amacıyla analiz eder. Veri madenciliği bu analizleri farklı veri madenciliği tekniklerinin kullanılması ile gerçekleştirmektedir.

Veri madenciliği teknikleri iki kategoride sınıflandırılabilir (Han ve Kamber, 2001):

i) Tanımlayıcı Veri madenciliği Teknikleri: Tanımlayıcı veri madenciliği teknikleri, veritabanı içerisinde yer alan verinin genel özelliklerinin açığa çıkartılmasını sağlamaktadır.

ii) Tahmin Edici Veri madenciliği Teknikleri: Tahmin edici Veri madenciliği teknikleri, mevcut veriden sonuç çıkarılmasını gerçekleştirerek tahmin yapılabilmesini sağlamaktadır.

Sınıflandırma ve tahmin veri analizinin iki formudur ve önemli veri sınıflarının tanımlanmasını veya gelecekteki verinin eğiliminin tahmin edilmesini, oluşturduğu modelleri kullanarak gerçekleştirmektedir.

4.1 Sınıflandırma

Veri madenciliği uygulamalarında önemli bir yer tutan sınıflandırma işlemi, yeni karşılaşılan girdinin özelliklerinin incelenip, bu girdinin daha önce belirlenmiş olan sınıflardan hangisine ait olduğunu belirlemeyi amaç edinen bir yöntemdir.

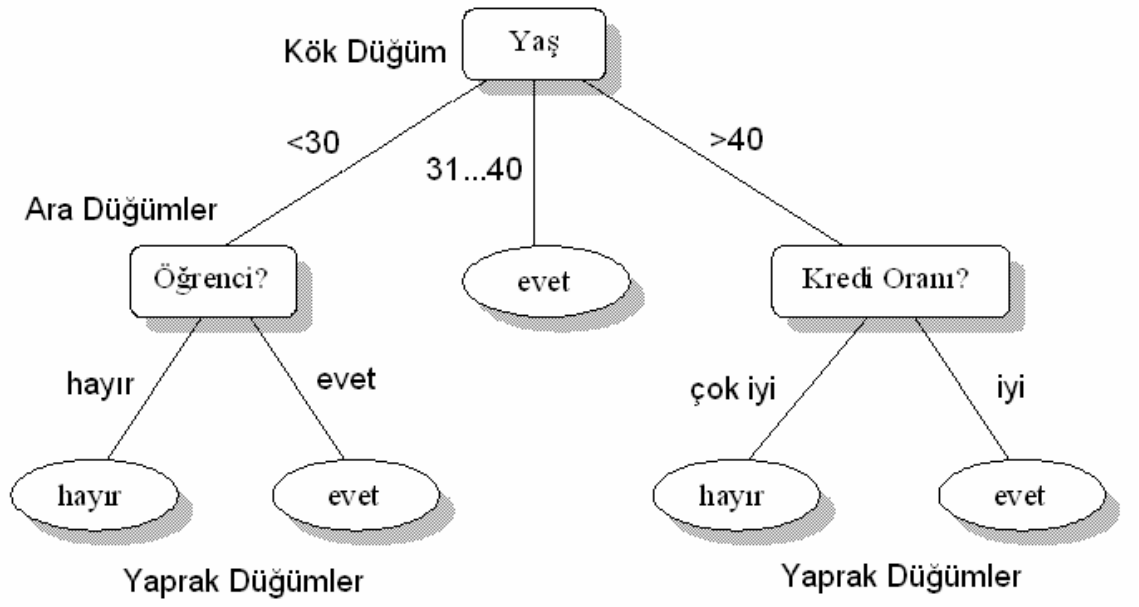
Sınıflandırmanın en önemli amacı, verinin sınıflandırılması sonucunda her sınıfta yer alan kişilerin, nesnelerin özelliklerinin ortaya çıkartılmasıdır. Örneğin, kredi kartı ile yapılan alış veriş sıklıklarına göre sınıflandırılan kredi kartı müşterileri, seyrek kullanıcı, orta sıklıkta kullanıcı ve sık kullanıcı olarak sınıflandırılabilir. Müşterilerin bu şekilde sınıflandırılması ile her bir sınıfın özellikleri ile tutum ve davranışlarını içeren bir model geliştirilebilir (Özmen, 2003).

Sınıflandırma aynı zamanda veri nesnelerinin sınıf etiketlerinin tahmin edilmesi için de kullanılabilir. Bununla birlikte birçok uygulamada karşımıza çıktığı gibi, kullanıcı bazı durumlarda sınıf etiketleri yerine kayıp veya mevcut olmayan veri değerlerini tahmin etmek isteyebilir. Özellikle sayısal değerlerin tahmin edilmesi söz konusu olduğunda, sayısal tahmin edici (numerical predictor) algoritmalardan olan Yapay Sinir Ağları ve Lineer Regresyon algoritmalarının kullanılması gerekmektedir (Han ve Kamber, 2001).

4.2 Karar Ağacı

Karar ağacı, ters çevrilmiş bir ağaç oluşturulacak şekilde bir akış diyagramının elde edildiği yapıdır. Bu yapı düğümlerden ve dallardan oluşmaktadır.

Karar ağacında en üstte yer alan düğüme kök düğüm, kendisine hiçbir bağlantının olmadığı düğüme yaprak düğüm ve bunların dışında kalan düğümlere ara düğüm denilmektedir. Dallar ise düğümler arasındaki bağlantının gerçekleştirildiği yapılardır. Genel olarak veri, karar ağacının düğümlerinde bulunur. Ara düğümlerde niteleyicilerin test değerleri, yaprak düğümlerde ise sınıf etiketleri bulunmaktadır. Karar ağacının dallarında düğümler arası geçişlerin sağlanması için gerekli olan koşullar yer almaktadır. Tipik bir karar ağacı yapısı Şekil 4.1'de gösterilmektedir.



Şekil 4.1 Elektronik ürünler satan bir mağazada, müşterilerin bilgisayar alıp almamasına göre karar ağacı yapısının oluşumu (Han ve Kamber, 2001).

Şekil 4.1 incelendiğinde; verinin düğümlerde, geçiş şartlarının ise dallarda bulunduğu görülmektedir.

Karar ağaçları genel olarak ikiye ayrılmaktadır (Berry ve Linoff, 2000):

i) Sınıflandırma Ağaçları: Sınıflandırma ağaçları kayıtları belli özelliklerine göre sınıflarına ayırır ve sınıflandırmanın anlamlılığı hakkında geribildirim sağlar. Sınıflandırma ağacı belli bir güven aralığında kaydın sınıf içerisinde yer aldığını

bilmemize imkan sağlar.

ii) Regresyon Ağaçları: Regresyon ağaçları sayısal değere sahip değişkenlerin değerini tahmin etmeye yarar. Regresyon karar ağacı ile sağlık sigortası olan bir kişinin sigortalı olduğu süre boyunca ne kadar sağlık harcaması yapacağı ile ilgili matematiksel bir denklem elde etmek mümkündür.

Karar ağacı oluşturulmasında aşağıdaki adımlar izlenir:

1. Bağımlı değişken ve söz konusu bağımlı değişkene etki ettiği düşünülen bağımsız değişkenler belirlenir. Bağımlı değişken aynı zamanda başlangıç noktasını oluşturan düğümü, bir başka deyişle kökü ifade etmektedir (Groth, 2000; Kovalerchuk ve Vityaev, 2000).

2. Bağımlı değişkene etki eden her bağımsız değişken incelenir. Bu aşamada karar ağacı tekrarlanan bölünmelere dayalı bir süreç izler. Karar ağaçlarında genellikle bölümlendirmenin anlamlılığı ki kare (χ^2) analizi ile tespit edilmektedir. Elde edilen ikili bölümlendirmeler arasında seçim yapılırken sınıflar arasındaki varyasyonun maksimize olduğu, sınıf içi varyasyonun ise minimize olduğu bölümlendirme tercih edilir (Berry ve Linoff, 2000; Groth, 2000).

3. Her değişken için yukarıda ele alınan işlemler tekrarlı bir şekilde gerçekleştirildikten sonra tahmin düzeyi en yüksek olan değişken seçilerek yaprak düğümleri oluşturmak için kullanılır (Groth, 2000).

Karar ağacı oluşturulurken ağaç gereksiz özellikler içererek büyüyebilir, gürültü sonucu sahte düğümler eklenebilir. Aşırı doyma olarak adlandırılan bu durumun üstesinden gelmek için ağacın budanması (pruning) gerekmektedir. Ağacın budanması sürecinde kullanılan 2 yaygın yaklaşım vardır:

i) Prepruning: Prepruning budama işleminde, Karar Ağacı yapısı oluşturulurken aynı zamanda budama işlemi de gerçekleştirilir.

ii) Postpruning: Postpruning budama işleminde, Karar Ağacı yapısı oluşturulduktan sonra budama işlemi gerçekleştirilir. Karar Ağacı algoritmaları arasında sık kullanılan algoritmalardan olan ID3 algoritması postpruning budama yöntemini kullanmaktadır.

Karar ağaçları yorumlanabilirliklerinin kolay olması, tanımlayıcı özelliklere sahip olması gibi

avantajlarından dolayı sınıflama modelleri içerisinde en yaygın kullanılan sınıflandırma yöntemidir (SPSS Inc.).

4.3 Yapay Sinir Ağları

Bilim adamlarının insan beyninin yapısını ve işleyişini taklit eden bir yapının oluşturulabileceğini düşünmeleri, Yapay Sinir Ağlarının (Neural Networks) ilk çıkış noktası olmuştur. Yapay sinir ağları sinir sisteminin işleyişini model olarak almaktadır. Yapay sinir ağları, deneyimleri kullanarak öğrenir ve girdi ile çıktılar arasındaki bilinmeyen ilişkileri belirler (Berson, 2000; Groth, 2000).

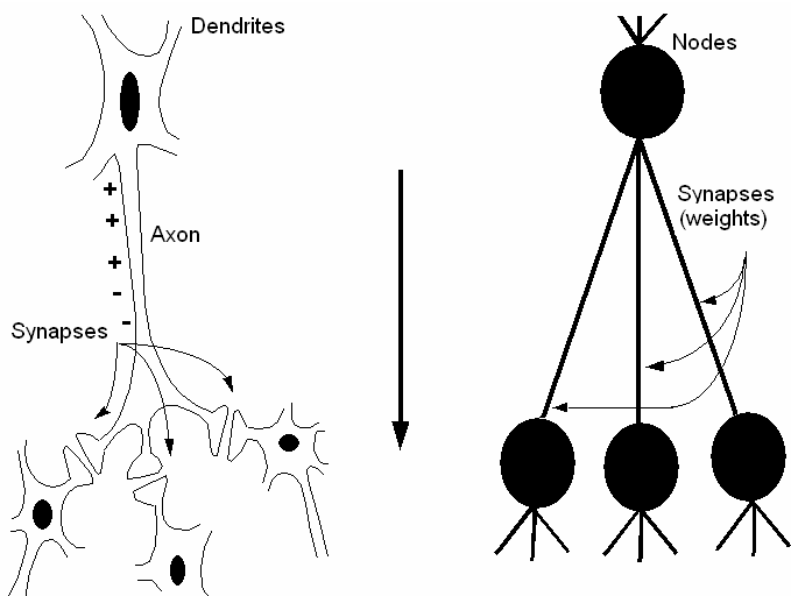
Yapay nöron ağlarının esin kaynağı olan insan nöron ağları, şu birimlerden oluşmaktadır:

- Akson (Axon): Akson, ucunda bulunan sinapsis denilen birimler ile hücre çekirdeğinden aldığı bilgiyi bir sonraki sinir hücresine dağıtmakla görevli birimlerdir.
- Dendrit (Dendrite): Diğer sinir hücrelerinden sinapsis vasıtasıyla getirilmiş olan sinyalleri hücre çekirdeğine iletmekten sorumludurlar.
- Sinapsis (Synapse): Akson ile dendrit arasındaki veri transferini, aksonlardan gelen toplam bilgiyi ön işlemlerden geçirerek, dendritlere iletmeyi sağlayarak gerçekleştiren birimdir. Ön işlem aşamasında gelen sinyalin belli bir eşik değerine indirgenmesi ile toplam sinyalin belli bir aralığa sığdırılması sağlanır ve bunun sonucu olarak toplam sinyal ile dendrit arasında bir korelasyon (ilişki) oluşturulur.
- Soma: Dendritler ile getirilen sinyallerin değerlendirildiği, gelen sinyal yeteri kadar baskın ise işleme sokulduğu, yeteri kadar baskın değilse işleme sokulmadığı biyolojik birimdir.

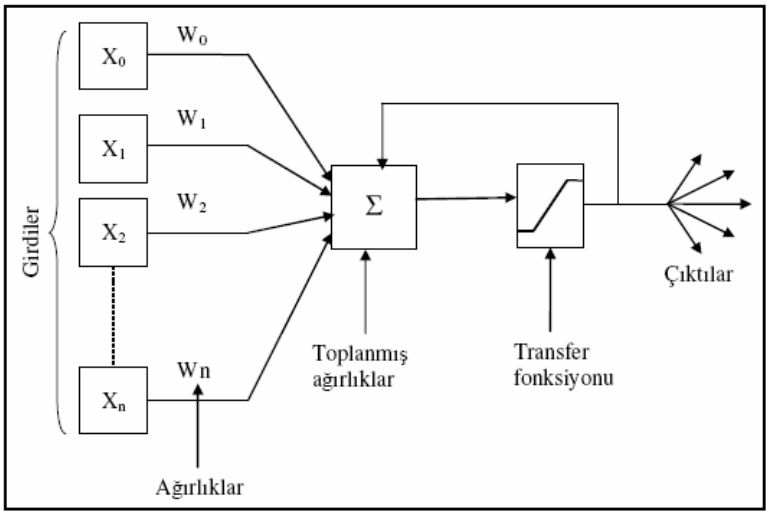
Yapay nöron ağı ile biyolojik nöron ağını oluşturan birimlerin, görevleri arasındaki benzerliklerine göre birebir eşlenmiş durumları Çizelge 4.1'de gösterilmektedir.

Çizelge 4.1 Biyolojik nöron ağı yapısına karşılık gelen yapay nöron ağı birimleri.

Biyolojik Nöron Ağı	**Yapay Nöron Ağı**
SOMA →	NÖRON
DENDRİT →	INPUT
SİNAPSİS →	WEIGHT
AKSON →	OUTPUT



Şekil 4.2 Biyolojik ve yapay nöron ağı yapısı.



Şekil 4.3 Yapay sinir hücresi modeli. X_i, giriş değerleri; W_i, bağlantı ağırlığıdır (Yazıcı vd., 2007).

Şekil 4.3'de görünen yapay sinir hücresinin dendritleri x_n ve her bir ağırlık katsayısı w_n ile gösterilmiştir. x_n girdi sinyallerini w_n ise o sinyallerin ağırlık katsayıları değerlerini taşımaktadır (Yazıcı vd., 2007).

Bir nöron diğer nöronlardan gelen sinyali birleştirir, dönüştürür ve sayısal bir sonuç elde eder. Gelen sinyal, şiddetine göre nöronlar tarafından sönümlendirilir ya da iletilir. Nöronlar tarafından, giriş değerlerinin her birini bir bağlantı ağırlığıyla çarpıp ağırlandırdıktan sonra bu

giriş değerlerini doğrusal olarak toplar ve eşik bu bilgiyi doğrusal veya doğrusal olmayan bir fonksiyonda işleyerek çıktı bilgisine dönüştürür. Bu çıktı bilgisi hücreye bağlantısı olan diğer nöronlar tarafından giriş bilgisi olarak kullanılır. Bu sürecin işleyişi sırasında öne çıkan temel sorun uygun ağırlık setinin belirlenmesidir. Uygun ağırlık setinin belirlenmesi için ağlar örnekler ile eğitilirler. İki öğrenme metodu kullanılmaktadır:

i) Eğiticili Öğrenme (Supervised Learning): Bu öğretme metodunda yapay sinir ağının dışarıdan etki ile eğitilmesi söz konusudur. Giriş değerlerinin ne tür bir çıktı vermesi gerektiği önceden bilinmektedir ve böylece yapay sinir ağı ağırlık değerleri bu korelasyona göre yeni değerler alabilmektedir. Bu öğrenme metodunda beklenilen çıktı değeri ile yapay sinir ağının verdiği çıktı değeri arasındaki hatanın ağırlıklara öğretilmesi hedeflenmektedir. Bu hata en az değerine ulaşıncaya kadar ağ nöronlar arasındaki ağırlıkları düzelterek iterasyona devam eder.

ii) Eğiticisiz Öğrenme (Unsupervised Learning): Bu öğrenme sürecinde yapay sinir ağı dışarıdan herhangi etki olmaksızın aldığı verileri bu veri grubuna uyumlu bir değer üretecek şekilde düzenlemektedir. Yapay sinir ağı bunu yapabilmek için ilk aldığı örnek/örnekleri bir sınıf olarak kabul eder. Daha sonra gelecek olan tüm girdi örüntüleri o sınıfa benzetilmeye çalışılır ve böylece tüm girdi örüntüleri kendi aralarında benzeyip benzememelerine göre ayırt edilir. Ancak bu sınıflandırma sürecinde hatalı cevaplar üretmek kaçınılmazdır. Yinede girdi örüntüleri sisteme bir çok defa öğrenim için beslenirse elde edilen yanılma payı %4 gibi makul bir değere düşebilecektir.

Yapay sinir ağlarının temel birimi olan düğüm (node), işlev olarak insan beynindeki nöronlara benzemektedir. Düğümler birbirlerine belirli ağırlıklar ile bağlıdır. Bir düğüm diğer bir düğümden belli miktar girdi alır ve bu girdiyi ağırlık değeri ile çarparak yeni bir toplam değer meydana getirir. Elde edilen değer aktivasyon fonksiyonu tarafından çıktı değeri almak için kullanılır ve sonraki düğüme gönderilir. Girdi olarak modele ilave edilen ifadeler sayısal değerlerin 0 ile 1 arasında normalize edilmiş, kategorik değerlerin ise 0 ile 1 değerleri kullanılarak kodlanmış halidir (Hair, 1998; Berson, 2000).

Yapay sinir ağında kullanılan temel aktivasyon fonksiyonları şunlardır:

1. Step fonksiyonu: Bu fonksiyon çıktı değeri olarak 0 veya 1 değeri üretmektedir.

$$Y_{step} = \begin{cases} 1 & eğer \quad x \geq 0 \\ 0 & eğer \quad aksi\,takdirde \end{cases}$$

2. Sign fonksiyonu: Bu fonksiyon çıktı değeri olarak 1 veya -1 değeri üretmektedir.

$$Y_{sign} = \begin{cases} 1 & eğer \quad x \geq 0 \\ -1 & eğer \quad aksi\,takdirde \end{cases}$$

3. Sigmoid fonksiyonu: Bu fonksiyonun matematiksel ifadesi;

$$Y_{sigmoid} = \frac{1}{1+e^{-x}} \tag{4.1}$$

şeklindedir.

4. Linear fonksiyon: Bu fonksiyonun matematiksel ifadesi;

$$Y_{linear} = x \tag{4.2}$$

şeklindedir.

Yapay sinir ağlarında, girdi sağlayan düğümlerden oluşan bir girdi katmanı ve çıktı düğümlerinden oluşan bir çıktı katmanı bulunmakta, bu katmanlar girdi ve çıktı değerleri arasında doğrudan ilişki bulunmadığı durumlarda iki değer arasındaki ilişkiyi açıklamak için kullanılmaktadırlar.

4.3.1 Çok Katmanlı Algılayıcı (Multilayer Perceptron)

En basit nörona Algılayıcı (Perceptron) denilmektedir. Algılayıcı, bugünkü yapay sinir ağları için önemli bir temel oluşturmaktadır. Algılayıcıda eşikleme fonksiyonu kavramı ile karşılaşılmakta, böylece eşiklemenin öğrenmedeki rolü kabul edilmektedir.

Algılayıcı modelinde ele alınan en önemli faktör eşik değeridir. Kullanılacak olan eşik değeri problemin karakteristiğine göre belirlenebilir. Bu şekilde Algılayıcı modeli ile daha başarılı bir modelleme gerçekleştirilebilmektedir.

Algılayıcı, iterasyon sayısının artırılması ile daha kararlı bir yapıya ulaşabilmektedir. Böylece A ve B gibi iki sınıf daha iyi bir şekilde konumlandırılabilecek aynı zamanda arada kalan kararsız bölge daha inceltilebilecektir.

Algılayıcı eğitim algoritması temel adımları şunlardır:

1. İlk değer belirleme: w girişlerin ağırlık değerlerini, θ ise sistemin eşik değerini temsil etmek üzere; $w_1, w_{2,.....,} w_n$ ve θ için [-0.5, +0.5] aralığında bir değer belirlenir.
2. İşlem: $x_1(p), x_2(p), ..., x_n(p)$ ilk değerleri için $Y_{d_1}(p), Y_{d_2}(p), ..., Y_{d_l}(p)$ istenen çıkışları elde edilir.

$$Y(p) = step[\sum_{i=1}^{n} x_i(p) * w_i(p) - \theta] \tag{4.3}$$

3. Ağırlıkların eğitilmesi: Δw_i i. nörona ait ağırlıkların değişimi, η öğrenme katsayısı (değeri genellikle 0 ile 1 arasındadır) olmak üzere ağırlıkların eğitilmesi;

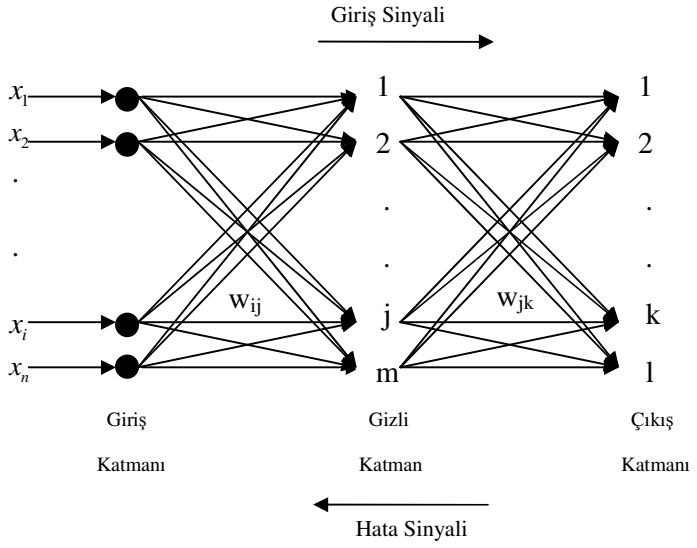
$$\Delta w_i = \mu * (\overbrace{y_d(p) - y(p))}^{hata} * x_i \tag{4.4}$$

ifadesi ile gerçekleştirilir.

4. İterasyon: Hata kalmayana veya en az değerine ulaşıncaya kadar 2. adımdan devam edilir.

Algılayıcı algoritması hataları öğrenek sistemi en az hataya indirgemeyi hedeflemektedir. Ne kadar hata yapılırsa o kadar iyi bir öğrenme söz konusudur.

Algılayıcı, lineer ayrılabilen iki sınıf üzerinde etkilidir. Aksi halde kullanılamaz. Lineer çözülemeyen problemler yapay ağa gizli katman katılarak çözülebilir. Bu şekildeki ağlara Çok Katmanlı (Multilayer) Ağlar denir. Gizli katman sayısı ne kadar arttırılırsa, o kadar karmaşık problemler çözülebilir.



Şekil 4.4 Çok katmanlı algılayıcı mimarisi.

Çok katmanlı algılayıcı eğitim algoritması şu adımları takip etmektedir:

1. İlk değer atama: Gizli katman giriş değerlerinin ağırlıkları ^{w}HO, gizli katman çıkış değerlerinin ağırlıkları ^{w}HO ile temsil edilmek üzere; ilk değer atamada $^{w}HO > {}^{w}HI$ şartı sağlanmalıdır.

2. İşlem: $x_1(p), x_2(p), ..., x_n(p)$ ilk değerleri için $Y_{d_1}(p), Y_{d_2}(p), ..., Y_{d_l}(p)$ istenen çıkışları elde edilir.

   i) Gizli katman çıkışı:

$$X_j(p) = sigmoid[\sum_{i=1}^{n} x_i(p) * w_{ij}(p) - \theta_j] \tag{4.5}$$

   n: j. nörona bağlı giriş sayısı.

   ii) Çıkış katmanının çıkışı (k. nöron için):

$$Y_k(p) = sigmoid[\sum_{j=1}^{m} x_{jk}(p) * w_{jk}(p) - \theta_k] \tag{4.6}$$

3. Ağırlıkların eğitilmesi (çıkıştan girişe doğru).

   i) Çıkış katmanı için hata (k. nöron için):

$$\delta_k(p) = y_k(p) * [1 - y_k(p)] * [\underbrace{y_{d,k} - y_k(p)}_{hata}] \tag{4.7}$$

$$\Delta w_{jk}(p) = \mu * y_j(p) * \delta_k(p) \tag{4.8}$$

$$w_{jk}(p+1) = w_{jk}(p) + \Delta w_{jk}(p) \tag{4.9}$$

   δ_k : k. nöron için w_{jk} bağlantısının hata hesabı.

   ii) Gizli katman için hata (j. nöron için):

$$\delta_j(p) = y_j(p) * [1 - y_j(p)] * \sum_{k=1}^{L} \delta_k(p) * w_{jk}(p) \tag{4.10}$$

$$\Delta w_{ij}(p) = \mu * x_i(p) * \delta_j(p) \tag{4.11}$$

$$w_{ij}(p+1) = w_{ij}(p) + \Delta w_{ij}(p) \tag{4.12}$$

δ_j : j. nöron için w_{ij} bağlantısının hata hesabı.

Yapay sinir ağları, verilerin modelde kullanılabilmesi için dönüştürülmeleri ve ortaya anlaşılması zor karmaşık modeller çıkarmaları gibi dezavantajlara sahiptir. Modelin sayısal veriler ile işlem yapması ve çıkış değerlerinin de sayısal değerler olması kategorik veriler kullanıldığında yorumlama açısından sorun oluşturmaktadır (Berson, 2000).

Yukarıda sayılan dezavantajlarına rağmen yapay sinir ağlarından, kredi kartı kullanımında dolandırıcılıkların saptanması ve kredi riskinin hesaplanması gibi konularda etkin bir şekilde yararlanılmaktadır. Bunun nedeni yapay sinir ağlarının geleneksel yöntemlere göre daha verimli ve aynı zamanda daha az maliyetli olmasıdır (Lanquillon, 1999; Ramamoorti, 1999).

4.4 Kümeleme

Kümeleme işlemi, heterojen yapıya sahip bir popülasyonu, daha homojen alt gruplara ya da kümelere ayırma işlemidir. Kümeleme analizinde hedeflenen, gruplar arasında maksimum heterojenlik, grup içerisinde ise maksimum homojenliğin sağlanmasıdır (Yeo, 2001).

Sınıflandırma ile kümeleme işlemleri birbirine sık karıştırılan yöntemlerdir. Sınıflandırma işleminde, yeni gelen her kayıt daha önce yapılan eğitim neticesinde elde edilen model ile belirlenmiş olan bir sınıfa atanmaktadır. Kümeleme işleminde ise önceden tanımlanmış sınıflar bulunmamaktadır.

Kümeleme işlemi çoğunlukla bir başka veri madenciliği işlemi için ilk adım olarak kullanılır.

Kümeleme analizi, birbirine benzer olan nesnelerin saptanması ve kümelerde toplanması amacıyla uygulanan bir istatistik analizidir. Başka bir deyişle, kümeleme analizi, x veri matrisinde yer alan ve doğal gruplamaları kesin olarak bilinmeyen birimleri, değişkenleri ya da birim değişkenleri birbirleri ile benzeri olan alt kümelere ayırmaya yardımcı olan yöntemler topluluğu olarak tanımlanabilir. Buna göre kümeleme analizi; birimleri, p değişkene göre hesaplanan ve benzerlik ölçüsü olarak kullanılan bazı ölçüler kullanarak homojen gruplara bölmek amacıyla kullanılır (Özdamar, 1999).

Başka bir deyişle kümeleme analizi sonucunda her nesne kendisine çok benzer diğer nesnelerle aynı küme içerisinde yer almaktadır. Buna göre kümeleme analizinde gruplar içi homojenlik ve gruplar arası heterojenlik düzeyi oldukça yüksek olmaktadır. Bu özelliği nedeniyle kümeleme analizi veri madenciliğinde uzun zamandır kullanılan yöntemlerden biridir ve özellikle pazarlama alanındaki müşteri segmentasyonu uygulamalarında

kullanılmaktadır (Hair, 1998; Berson, 2000).

Kümeleme analizi, temel olarak dört değişik amaca yönelik olarak uygulanan bir yöntemdir (Özdamar, 1999):

i) n sayıda durumu, nesneyi, oluşumu (phenemona) p değişkene göre saptanan özelliklerine göre olabildiğince kendi içinde türdeş (homojen) ve kendi aralarında farklı (heterojen) alt gruplara (kümelere) ayırmak,

ii) p sayıda değişkeni n sayıda birimde saptanan değerlere göre ortak özellikleri açıkladığı varsayılan alt kümelere ayırmak ve ortak faktör yapıları ortaya koymak,

iii) Hem birimleri hem de değişkenleri birlikte ele alarak ortak n birimi p değişkene göre ortak özellikli alt kümelere ayırmak,

iv) Birimleri, p değişkene göre saptanan değerlere göre, izledikleri biyolojik ve tipolojik sınıflamayı ortaya koymak.

Kümeleme analizinin uygulama aşamaları aşağıdaki gibi verilebilir (Özdamar, 1999):

i) Birim ya da değişkenlerin doğal gruplamaları hakkında kesin bilgilerin bulunmadığı popülasyonlardan alınan n sayıda birimin p sayıda değişkenine ilişkin gözlemlerin elde edilmesi (veri matrisinin belirlenmesi).

ii) Birimlerin/değişkenlerin birbirleri ile olan benzerliklerini (similarity) ya da farklılıklarını (dissimilarity) gösteren uygun bir benzerlik ölçüsü ile birimlerin/değişkenlerin birbirlerine uzaklıklarının hesaplanması (benzerlik ya da farklılık matrisinin belirlenmesi).

iii) Uygun kümeleme yöntemi yardımı ile benzerlik/farklılık matrislerine göre birimlerin/değişkenlerin uygun sayıda kümelere ayrılması.

iv) Elde edilen kümelerin yorumlanması ve bu kümeleme yapısına dayalı olarak kurulan hipotezlerin doğrulanması için gerekli analitik yöntemlerin uygulanması.

Özetle, kümeleme analizi çok sayıda değişik işlevi yerine getiren yöntemler topluluğudur. Bu nedenle kümeleme analizinde farklı amaçlar için farklı yöntemler uygulanmaktadır. Ayrıca değişkenlerin ölçü birimlerinin ve ölçümleme tekniklerinin farklı olmasından dolayı birimlerin benzerliklerinin ortaya konmasında da değişik ölçüler kullanılmaktadır.

Kümeleme analizi doğal sınıflamaları hakkında açık bilgi bulunmayan durumlarda, popülasyona ilişkin tahminlerin yapılmasında yararlanılan bir yöntemler topluluğudur. Bu nedenle doğal gruplamaları açıkça bilinen toplumlarda alt kümelerin irdelenmesi ayırma analizi ile yapılır. Alt popülasyon tanımlamaları açıkça yapılamamış ya da ayrı ayrı

popülasyonlar oldukları kesin bilinmeyen karma toplumları birbirinden ayırmak, yeni tanımlamalar yapmak, birimler için yeni prototipler belirlemek, popülasyon ya da alt popülasyon profilleri tanımlamak amacıyla kümeleme analizinden yararlanılmaktadır.

Popülasyon tanımlamaları yapılmamış, toplum profilleri açıkça ortaya konmamış toplumlarda örnek yapılar belirlemek, birimler için prototipler belirlemek, sınıflandırma grafikleri ortaya koymak için kümeleme analizine başvurmak gerekmektedir.

Kümeleme analizinde birimlerin p değişkene göre birbirleri arasındaki uzaklıklarını hesaplamak için çok çeşitli uzaklık ölçü birimleri ileri sürülmüştür. Uzaklık ölçüleri ya da benzerlik ölçüleri veri matrisinde yer alan değişkenlerin ölçü birimlerine göre de farklılık göstermektedir. Eğer değişkenler oransal ya da aralıklı ölçekle elde edilmiş değerler ise uzaklık ya da ilişki türü ölçülerinden yararlanılır. Ölçümler sayısal değerler olarak yapılmış ise tercih edilen ölçüler Ki-Kare Uzaklık Ölçüsü ya da Phi-Kare'dir. Eğer ikili (binary) gözlemlere göre ölçümler yapılmış ise birimler arasındaki benzerlikleri belirlemede Öklid, Kare Öklid, Boyut Farkı gibi benzerlik ya da farklılık ölçütlerinden faydalanılmaktadır (Özdamar, 1999).

Genelde uzaklık ölçüleri, doğrudan birim ya da değişkenlerin kümelenmesinde kullanılabileceği gibi birim ya da değişkenler arasındaki benzerlik ya da farklılıkları hesaplamakta da kullanılabilir. Kümeleme analizinde, birimler arasındaki uzaklıkların hesaplanmasında sıklıkla kullanılan ölçütler şunlardır:

i) Öklid Uzaklığı: Birimler arasındaki uzaklığın hesaplanmasında kullanılan yöntemler arasında en çok tercih edilen uzaklık ölçütüdür. n*p boyutlu bir veri matrisinde i. ve j. birimler (gözlemler, nesneler) arasındaki Öklid Uzaklığı şu şekilde hesaplanır:

$$d(i,j) = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2} \qquad (4.13)$$

Burada i=1,2,….,n; j=1,2,…,n ve k=1,2,...,p'dir. n birim sayısı, p değişken sayısı, x_{ik} i. birimin k. birime olan uzaklığı, x_{jk} ise j.birimin k.birime olan uzaklığıdır.

ii) Pearson Uzaklığı: Öklid uzaklığının değişkenin varyansına oranlanmış uzaklığıdır.

$$d_p(i,j) = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2 / s_k^2} \qquad (4.14)$$

(4.15) eşitliğinde yer alan s_k^2 değişkenin varyansını belirtmektedir.

iii) Manhattan Uzaklığı: Birimler arasındaki mutlak uzaklıkların toplamının alınması ile hesaplanan bir uzaklık ifadesidir.

$$d_M(i,j) = \sum_{k=1}^{p} | x_{ik} - x_{jk} | \tag{4.15}$$

Veri matrisinde değişkenlerin ortalama ve varyansları birbirinden çok farklı olduklarında büyük ortalama varyansa sahip değişkenler diğer değişkenlerin rollerini önemli oranda baskılamakta ve etkilemektedir. Bazen değişkenlerin aşırı uçlarda yer alan değerleri kümeleme üzerinde olumsuz etkilerde bulunmaktadır. Bu gibi durumlarda verilerin standardize ya da belirli aralıklarda gözlenen değerlere dönüştürülmesi uygun olmaktadır (Özdamar, 1999).

Verilerin standardize edilmesi ya da belirli aralıklara dönüştürülmesinde kullanılan çeşitli yöntemler bulunmaktadır (Özdamar, 1999):

i) z Skorlarına Dönüştürme: En çok tercih edilen dönüştürme yöntemidir. Oransal ya da aralıklı ölçekle elde edilen, normal dağılım gösterdiği varsayılan verilere uygulanır ve değerler,

$$z_i = \frac{x_i - \overline{x}}{s} \tag{4.16}$$

biçiminde z skorlarına dönüştürülür. x_i verideki i. kayıdın değerini, $\overline{x}$ verideki kayıtların aritmetik ortalamasını, s ise kayıtların standart sapmasını belirtmektedir.

ii) $-1 \le x \le +1$ Aralığına Dönüştürme: Heterojen yapıda değerlerin ve aşırı uçlarda değerlerin yer aldığı durumlarda tercih edilen bir dönüştürme yöntemidir. $x_{\max}$ dizideki en büyük değer olmak üzere dönüştürme;

$$x_i = \frac{x_i}{x_{\max}} \tag{4.17}$$

biçiminde yapılır.

iii) $0 \le x \le +1$ Aralığına Dönüştürme: Heterojen yapıda değerlerin ve aşırı uçlarda değerlerin yer aldığı durumlarda değerleri pozitif ve 0-1 aralığında değişecek biçime dönüştürmek için tercih edilen bir dönüştürme yöntemidir.

Dizide en büyük değer $X_{\max}$ ve en küçük değer $X_{\min}$ olmak üzere dönüştürme;

$$R = X_{\max} - X_{\min} \tag{4.18}$$

$$x_i = \frac{x_i - x_{\min}}{R} \tag{4.19}$$

biçiminde yapılır.

Kümeleme analizi, uzaklık matrisi ya da benzerlik matrisinden yararlanarak birimler ya da değişkenleri kendi içinde homojen ve kendi aralarında heterojen gruplar oluşturmayı sağlayan yöntemler aracılığıyla gerçekleştirilir. Söz konusu yöntemler, birim ya da değişkenleri uygun gruplara ayırırken, grupları belirlemede (kümelemede) izledikleri yaklaşımlara göre iki temel gruba ayrılırlar: Hiyerarşik ve hiyerarşik olmayan kümeleme analizi (Özdamar, 1999).

4.4.1 Hiyerarşik Kümeleme Analizi

Hiyerarşik yöntemler sıra gözetilerek yapılan işlemleri içermektedir. Böylece başlangıçta belli hesaplamalar yapılmadan bir sonraki adıma geçilemez. Hiyerarşik olmayan yöntemlerde ise böyle bir sıralama düzenine gerek kalmadan işlemler gerçekleştirilmektedir. Buna göre hiyerarşik kümeleme analizinde kümeler küçükten büyüğe doğru belli bir hiyerarşi içerisinde oluşturulmaktadır. Bunun temelinde tek bir mutlak doğru sonucun bulunmadığı varsayımı yatmaktadır. Bu nedenle amaç az ya da fazla sayıda kümelerin elde edilmesi olmaktadır (Manly, 1989; Berson, 2000).

Hiyerarşik kümeleme, kayıtları kümelere ayırırken iki farklı yöntem izlemektedir:

i) Birleştirici (Agglomerative) Yöntem: Bu yöntem her kaydı ayrı bir küme olarak kabul eder ve bu kümelerden(kayıtlardan) tek büyük bir küme oluşuncaya kadar kümeleri birleştirir. Yöntem başlangıçta sadece bir nesneye sahip kümeler ile başka deyişle her kaydı bir küme kabul ederek analize başlar. Daha sonra söz konusu kümeleri benzerliklerine göre birleştirir (Johnson ve Wichern, 1988; Berry ve Linoff, 2000).

ii) Bölünebilir (Divisive) Yöntem: Bölünebilir yöntem birleştirici yöntemin tam tersi şekilde işler. Başlangıçta tüm kayıtlar tek büyük bir kümenin bir parçası olarak kabul edilir ve daha sonra bu küme iki veya üç alt kümeye bölünür. Her alt küme birbirine benzemeyen alt kümelere bölünerek işlem devam eder. Sonuçta oluşturulan kümeler içerisinden en iyi olanları seçilir (Berry ve Linoff, 2000).

4.4.2 Hiyerarşik Olmayan Kümeleme Analizi

Hiyerarşik olmayan kümeleme yöntemi, birimlerin kendi içinde homojen ve kendi aralarında heterojen olan kümelere ayrılmasını hedefleyen ve prototip kümeler aracılığı ile alt popülasyonların parametre tahminlerini yapmayı (grup ya da küme ortalama vektörleri ve kovaryans matrisleri) amaçlayan yöntemlerdir. Aşamalı kümelemede hem birimler hem de değişkenler birbirleri ile değişik benzerlik düzeylerinde kümeler oluştururken, aşamalı olmayan yöntemlerde birimlerin uygun oldukları kümelerde toplanmaları ve n birimin k sayıda kümeye parçalanması hedeflenmektedir.

Hiyerarşik olmayan kümeleme yöntemlerinde n birimin k kümeye parçalanması rasgele ya da verilerin incelenmesi ile gelişigüzel yapılabilir. Birimlerin ayrılabilecekleri küme sayısı belirlendikten sonra kümeler için belirlenen küme belirleme kriterlerine göre birimlerin hangi kümelere girebileceklerine karar verilir ve atama işlemleri yapılır (Özdamar, 1999).

Hiyerarşik olmayan kümeleme analizinde ağaç benzeri yapılar oluşturulmaz. Hiyerarşik kümelemeden önemli bir farkı ise araştırmacının analiz başında, olmasını istediği küme sayısını belirtmesidir. Analiz, rasgele belirlediği bir noktadan kümeleme çalışmalarına başlar ve bu noktadan önceden belirlenmiş bir mesafe kadar yakınında bulunan kayıtları bu küme içerisine alır. Daha sonra yeni bir nokta belirleyerek kümeleme çalışmalarına belirlenmiş sayıda küme oluşturuncaya kadar devam eder. İşlemler tamamlandıktan sonra herhangi bir kümedeki kayıt bir başka kümedeki kayıtlara benzer ise o kümeye aktarılarak yer değişimleri gerçekleştirilir (Hair, 1998).

Hiyerarşik olmayan kümeleme algoritmaları arasında en çok öne çıkanı K-Means algoritmasıdır.

4.4.2.1 K-Means Algoritması

K-Means algoritmasında veri seti önceden belirlenmiş sayıda (k) kümeye bölünmekte ve her bir küme için bir başlangıç merkez noktası (centroid) belirlenmektedir. Bu işlemden sonra nesneler yakınlarındaki noktalara atanmakta ve her atamadan sonra küme merkezleri yeniden hesaplanmaktadır. Bu işlemler merkezi noktalar değişmeyene kadar sürdürülmektedir. Genellikle söz konusu noktalar rasgele seçilmekte ve bu durum tatminkar sonuçlar alınmamasına yol açmaktadır. Bunun yerine merkezi başlangıç noktalarının veri setinin sıkışık bölgelerinden seçilmesi önerilmektedir. Böylece noktaların birbiriyle daha iyi bir şekilde ayrışmaları sağlanabilir. Bu durum aynı zamanda bir kümeden birden fazla merkezin

ortaya çıkmasına engel olur (Berry ve Linoff, 2000).

K-Means yöntemi aşağıda belirtilen adımları kullanarak kümeleme işlemini gerçekleştirir (Özdamar, 1999).

i) Verilerden elde edilecek bilgilere göre ilk nokta çekirdek nokta olarak ele alınır ve bu noktaların her birinin p değişken değerleri birer küme ortalama vektörü olarak kabul edilir. Küme ortalama vektöründen her bir birimin uzaklıkları hesaplanır.

ii) Geriye kalan n-k birim en yakın ortalama vektörlü kümeye atanır. Her atamadan sonra oluşan kümenin ortalama vektörü yeniden hesaplanır. En yüksek benzerliğe sahip birimler bir araya getirilir.

iii) Eldeki veri, küme içi varyansın minimum ve kümeler arası varyansın maksimum olduğu kümeleme yapısına ulaşıncaya kadar k adet kümeye atanmaya devam eder.

4.5 Regresyon Analizi

Regresyon analizi istatistikte tahmin amacıyla kullanılan bir yöntemdir ve değişik türleri bulunmaktadır. Bunların içerisinde en temeli lineer regresyon analizidir. Basit regresyon analizi Y bağımlı değişkeninin tek bir X bağımsız değişkeni ile arasındaki ilişkinin doğrusal fonksiyon ile ifade edilmesidir (Orhunbilge, 1996).

Regresyon analizindeki bağımlı değişken; değeri başka değişkenler tarafından etkilenen ve diğer değişkenlerin değeri değiştiğinde bu değişimden etkilenen değişkendir. Bağımsız değişken ise değeri rasgele koşullara göre belirlenen, bağımsız olarak değişim gösteren ve başka değişkenlerin değişimi üzerinde etkide bulunan, değişkenlere bağımsız ya da açıklayıcı değişken adı verilir (Özdamar, 1999).

Regresyon analizi iş dünyasında sık karşılanılan bir durum olan evet/hayır şeklinde yargılara varmak için kullanılmak istendiğinde, olası iki sonuç ortaya çıkabileceğinden ve bu durumun bir doğru ile açıklanması mümkün olmadığından lineer regresyon kullanılamaz. Bu durumlarda regresyon analizinin bir başka türü olan lojistik regresyon yöntemi kullanılmaktadır (Berson, 2000).

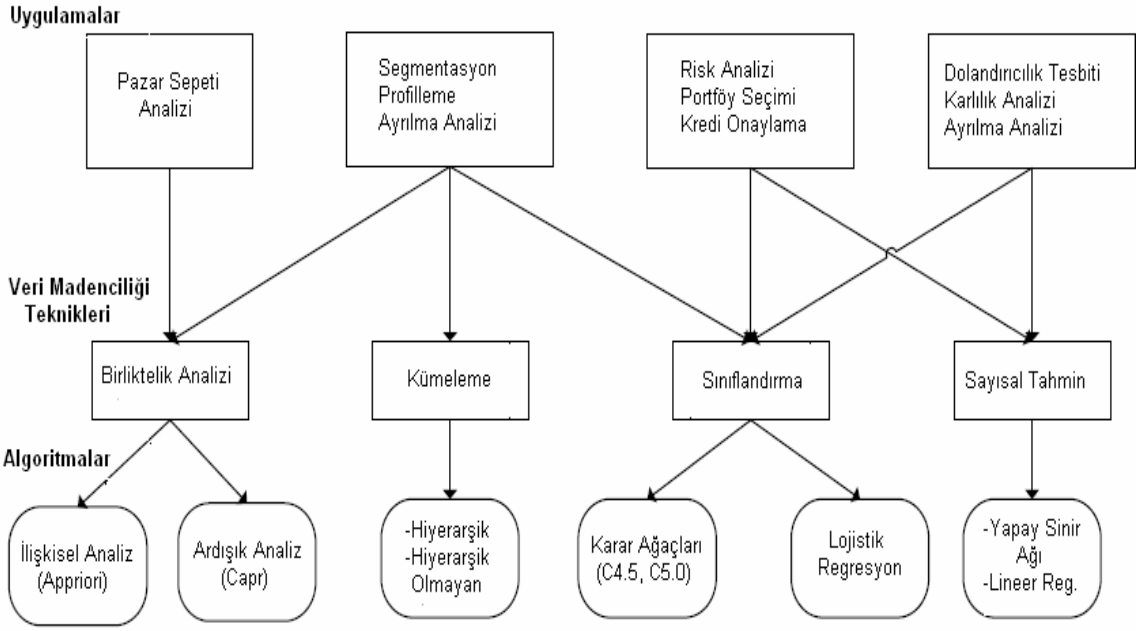
Lojistik regresyon, çoklu regresyondan farklı olarak doğrudan bir olayın gerçekleşme ihtimalini tahmin etmektedir. Olasılık değerleri 0 ile 1 arasında herhangi bir değer olabilmektedir. Sıfır ve bir arasında bir ilişki kurabilmek için lojistik regresyon bağımlı ve bağımsız değişken arasındaki ilişkinin S şeklinde olduğunu varsayar. Bağımsız değişkenin çok düşük değerlerinde olasılık sıfıra yaklaşmaktadır. Bağımsız değişken arttıkça olasılık eğri

üzerinde artmaktadır ancak sonra eğim biri aşmamak üzere azalmaya başlamaktadır (Hair, 1998).

4.6 Birliktelik Analizi (Association Analysis)

İlişki kurma daha çok hangi ürünlerin hangi ürünler ile daha çok satıldığı ile ilgili bilgi vermek amacıyla kullanılan bir yöntemdir. Özellikle süpermarketlerdeki alışveriş kayıtlarının incelenerek hangi ürünlerin beraber satıldıklarının belirlenmesinde yaygın olarak kullanılmaktadır. Bu şekilde elde edilen veriler ışığında süpermarketlerdeki raf düzenleri beraber satılan ürünlerin birlikte gözükmesini sağlayacak şekilde yeniden oluşturulabilmektedir.

Her bir veri madenciliği tekniğinin kullanıldığı belli başlı algoritmalar ve bu tekniklerin uygulama alanları Şekil 4.4'de özetlenmiştir.



Şekil 4.4 Veri madenciliğinin uygulama alanları, teknikleri ve algoritmaları (Musaoğlu, 2003).

5. VERİ MADENCİLİĞİ UYGULAMA SÜRECİ

5.1 Verinin Tanıtılması ve Verinin Önişleme Süreci

5.1.1 Veri Madenciliği Sürecinde Kullanılacak Verinin Tanıtılması

Uygulamada kullanılacak olan veri, veri madenciliği sürecinde kullanmak üzere veri arayan araştırmacılar için internet ortamına koyulan bir Çek Cumhuriyeti bankasına ait veritabanından elde edilmiştir. Elde edilen veritabanı, her biri farklı ASCII dosyalarda yer alan 8 adet tablodan oluşmaktadır.

Account Tablosu: Banka müşterilerinin sahip olduğu hesaplar ve bu hesapların özellikleri hakkında bilgi vermeyi amaçlayan 4500 kayıtlık bir tablodur. Bu tabloda yer alan tanımlayıcılar aşağıdaki gibidir.

1. Account_id: Müşterilerin işlem yaptığı hesapların, hesap numaraları bilgisini tamsayı formatında tutan kayıttır.

2. District_id: Müşteriye ait hesabın açıldığı bankanın bulunduğu bölge ile ilgili bilgiyi, tamsayı formatında tutan kayıttır.

3. Date: Müşteriye ait hesabın açılma tarihi bilgisini tutan kayıttır. Bu kayıt YYMMDD formatındadır.

4. Frequency: Müşteriye ait hesabın kullanılma sıklığı ile ilgili bilgi veren kayıttır. Hesabın kullanılma sıklığı aylık kullanım, haftalık kullanım ve haftalıktan az kullanım olmak üzere üçe ayrılmıştır.

    - Poplatek Mesicne: Aylık kullanım.
    - Poplatek Tydne: Haftalık kullanım.
    - Poplatek Po Obratu: Haftalıktan az kullanım.

Client Tablosu: Bankada hesabı bulunan veya bu hesap üzerinde işlem yapabilme hakkına sahip olan müşteriler ile ilgili kayıtların tutulduğu tablodur. Tablo 5369 kayıttan oluşmuştur.

1.Client_id: Bankada hesabı bulunan müşterilerin, müşteri numarası bilgisini tamsayı formatında tutan kayıttır.

2.Birth number: Müşterilerin doğum tarihleri ve cinsiyeti bilgilerinin bir arada tutulduğu kayıttır. Bu kayıtta doğum tarihinin ay hanesine 50 eklenerek müşterinin bayan olduğu

vurgulanmak istenmiştir.

3. District_id: Müşterinin yaşadığı bölge ile ilgili bilginin tamsayı formatında tutulduğu kayıttır.

Disposition Tablosu: Bu tablo, müşteri bilgileri ile hesap bilgilerinin birbirine bağlanmasını kolaylaştırmak amacıyla oluşturulmuş bir tablodur. Tabloda 5369 kayıt bulunmaktadır.

1. Disp_id: Disposition tablosundaki verileri tanımlayan tamsayı formatında bilgi tutan kayıttır.

2. Account_id: Müşterilerin işlem yaptıkları hesapların, hesap numaralarını tamsayı formatında tutan kayıttır.

3. Type: Müşterilerin, sahip olduğu hesaplar üzerindeki yetkilerinin tanımlandığı kayıttır. Müşteriler, hesap sahipleri(OWNER) ve hesap üzerinde kısıtlı işlem yapabilenler(DISPONENT) olmak üzere iki gruba ayrılmıştır.

    - OWNER: Hesap sahibi olarak tanımlanan bu müşteriler, bankanın müşterilerine sunduğu bütün imkanlardan faydalanabilmektedir.
    - DISPONENT: Bankanın müşterilerine sunduğu imkanların bir kısmından yararlanma hakkına sahip olan bu müşteri grubu, hesaptan otomatik ödeme yapma ve bankadan kredi çekme talebinde bulunma işlemlerini gerçekleştirememektedir.

Permanent Order: Müşterinin hesabından yapılmasını istediği otomatik ödemeler ile ilgili bilgileri tutan tablodur. Bu tablo 6471 kayıttan oluşmaktadır.

1. order_id: Müşterilerin yapmış olduğu otomatik ödemeleri tamsayı formatında numaralandıran kayıttır.
2. account_id: Müşterinin otomatik ödemesini hangi hesaptan yaptığını tamsayı formatında gösteren kayıttır.
3. bank_to: Müşterinin otomatik ödemesini hangi bankaya yaptığını gösteren kayıttır. Ödemelerin yapıldığı bankaların isimleri iki harf (MN, YZ gibi) ile kodlanmıştır.
4. account_to: Müşterinin otomatik ödemesini farklı bir bankanın hangi hesabına yaptığı bilgisinin tamsayı formatında tutulduğu kayıttır.
5. amount: Müşterinin yapmış olduğu otomatik ödemenin miktarı ile ilgili bilgilerin tamsayı formatında tutulduğu kayıttır.

6. k_symbol: Başka bankaya yapılmış olan otomatik ödemelerin karakteristik özellikleri ile ilgili bilgi veren kayıtlar tutulmaktadır.
    - POJISTNE: Sigorta ödemesi.
    - SIPO: Ev giderleri (elektrik, su, telefon vb) ödemeleri.
    - LEASING: Lizing (finansal kiralama) ödemesi.
    - UVER: Kredi ödemesi.

Transaction Tablosu: Müşterilerin 01.01.1983 ile 31.12.1998 tarihleri arasında yapmış olduğu banka işlemleri ile ilgili bilgileri tutan tablodur. Müşterilerin hesap bilgilerinin tanımlandığı 8 Çizelge arasında en fazla kayıda sahip (1.056.320 kayıt) tablodur.

1. trans_id: Müşterinin yapmış olduğu işlemi tamsayı formatında numaralandıran kayıttır.
2. account_id: Yapılan işlemin hangi hesaptan yapıldığı ile ilgili bilgiyi tamsayı formatında tutan kayıttır.
3. date: İşlemin hangi tarihte yapıldığı bilgisini YYMMDD formatında tutan kayıttır.
4. type: Yapılan işlemin hesaptan para çıkışı olup olmadığı ile ilgili bilgiyi tutan kayıttır. Bu bilgiler iki ifade ile tanımlanmıştır:
    - PRIJEM: Hesaba para girişini ifade eder.
    - VYDAJ: Hesaptan para çıkışını ifade eder.
5. operation: Müşterinin yapmış olduğu işlemin modunu tanımlar.
    - VYBER KARTOU: Kredi kartından para çekilme işleminin yapıldığını ifade eder.
    - VKLAD: Hesaba para yatırılması işlemini ifade eder.
    - PREVOD Z UCTU: Başka bankadan hesaba para aktarılma işlemini ifade eder.
    - VYBER: Hesaptan para çekilme işlemini ifade eder.
    - PREVOD NA UCET: Başka bankaya para havale edilmesi işlemini ifade eder.
6. amount: Yapılan işlemlerin miktarı ile ilgili bilgiyi tamsayı formatında tutan kayıttır.
7. balance: Yapılan işlem sonrasında hesapta kalan para miktarı bilgisini tamsayı formatında tutan kayıttır.
8. k_symbol: Yapılan işlemin karakteristik özellikleri ile bilgileri tutan kayıttır.
    - POJISTNE: Müşterinin sigorta ödemesi yapmış olduğunu belirtir.
    - SLUZBY: Müşterinin yaptığı hesap işlemleri için bankadan ücret karşılığı

talep ettiği hesap özetini tanımlar.

- UROK: Müşterinin hesabındaki para için aldığı aylık faiz bilgisini belirtir.
- SANKC.UROK: Müşterinin hesabının eksiye düşmesi nedeniyle, bankanın müşteriye belli faiz oranı karşılığında para verdiği bilgisini tutar.
- SIPO: Müşterinin ev giderleri (elektrik, su, telefon vb) ödemesi yaptığını belirtir.
- DUCHOD: Müşterinin hesabına emekli maaşı ödemesi yapıldığını belirtir.
- UVER: Müşterinin almış olduğu kredinin taksitle geri ödendiği bilgisini belirtir.

9. bank: Müşterinin işlemini hangi bankaya yaptığını tamsayı formatında gösteren kayıttır.
10. account: Müşterinin işlemini bankanın hangi hesabına yaptığını tamsayı formatında gösteren kayıttır.

Loan Tablosu: Bankanın sunduğu kredi imkanından yararlanan müşterilerin bilgilerinin tutulduğu, 682 kayıttan oluşan bir tablodur.

1. loan_id: Alınan kredinin kredi numarası ile ilgili bilgiyi tamsayı formatında tutan kayıttır.
2. account_id: Krediyi alan müşterinin hesap numarası bilgisinin tamsayı formatında tutulduğu kayıttır.
3. date: Kredinin alınma tarihi (YYMMDD formatında) bilgisinin tutulduğu kayıttır.
4. amount: Alınan kredinin miktarının tamsayı formatında tutulduğu kayıttır.
5. duration: Alınan kredinin ay bazında geri ödenme süresi bilgisini tamsayı formatında tutan kayıttır.
6. payments: Kredinin geri ödenmesi aşamasında yapılacak aylık geri ödemelerin miktarı bilgisini tamsayı formatında tutan kayıttır.
7. status: Kredinin geri ödenmesi durumu ile ilgili bilgilerin tutulduğu kayıttır.
    - A: Kredi sözleşmesinde belirtilen süre içerisinde kredinin geri ödenmesi tamamlandı.
    - B: Kredi sözleşmesinde belirtilen süre içerisinde kredinin geri ödenmesi tamamlanamadı.
    - C: Kredi geri ödeme süreci devam etmekte ve ödemelerde sorun yok.
    - D: Kredi geri ödeme süreci devam etmekte ve ödemelerde sorun var.

Credit Card Tablosu: Bu tabloda müşterilerin sahip olduğu kredi kartları ile ilgili bilgilerin

tutulduğu kayıtlar bulunmaktadır. Kredi kartı tablosunda 892 kayıt bulunmaktadır.

1. card_id: Müşterinin sahip olduğu kredi kartının numarası bilgisini tamsayı formatında tutan kayıttır.
2. disp_id: Kredi kartına sahip müşteri ile müşterinin hesabı arasında, disp tablosunu kullanarak köprü kuran kayıttır. Kayıtlar tamsayı formatındadır.
3. type: Alınan kredi kartının tipi ile ilgili bilgilerin tutulduğu kayıttır. Alınan kredi kartı tipleri “junior”, ”classic” ve “gold” kart olmak üzere üçe ayrılmaktadır.
4. issued: Alınan kredi kartının basım tarihi (YYMMDD formatında) bilgisini tutan kayıttır.

Demographic Tablosu: Müşterilerin yaşadığı ve hesaplarının bulunduğu banka şubelerinin bulunduğu yerleşim yerleri ile ilgili demografik bilgilerin tutulduğu tablodur. Demographic tablosunda 77 kayıt bulunmaktadır.

1. A1(district_id): Müşterinin yaşadığı veya hesabını açtırdığı bankanın şubesinin bulunduğu bölgeyi tamsayı formatında numaralandıran kayıttır.
2. A2: Müşterinin yaşadığı veya bankanın şubesinin bulunduğu bölgenin ad bilgisinin tutulduğu kayıttır. Bu kayıtta 77 farklı bölge tanımlanmaktadır.
3. A3: Bölgenin bağlı bulunduğu şehrin adı bilgisini tutan kayıttır. Bu kayıtta 8 farklı şehir vardır.
4. A4: Şehirde yaşayan kişilerin sayısını tamsayı formatında tutan kayıttır.
5. A5: Şehirde yer alan ve nüfusu 499’dan küçük belediyelerin sayısını tamsayı formatında veren kayıttır.
6. A6: Şehirde yer alan ve nüfusu 500 ile 1999 arasında olan belediyelerin sayısını tamsayı formatında veren kayıttır.
7. A7: Şehirde yer alan ve nüfusu 2000 ile 9999 arasında olan belediyelerin sayısını tamsayı formatında veren kayıttır.
8. A8: Şehirde yer alan ve nüfusu 10000’den fazla olan belediyelerin sayısını tamsayı formatında veren kayıttır.
9. A9: Şehirde yer alan belediyelerin toplam sayısı ile ilgili bilgiyi tamsayı formatında veren kayıttır.
10. A10: Müşterinin yaşadığı şehirdeki, şehirleşme oranı bilgisini tamsayı formatında veren kayıttır.
11. A11: Müşterinin yaşadığı şehirdeki, ortalama maaş bilgisinin tamsayı formatında tutulduğu kayıttır.

12. A12: Müşterinin yaşadığı şehirde, 1995 yılına ait işsizlik oranı bilgisinin tamsayı formatında tutulduğu kayıttır.
13. A13: Müşterinin yaşadığı şehirde, 1996 yılına ait işsizlik oranı bilgisinin tamsayı formatında tutulduğu kayıttır.
14. A14: Şehirde yaşayan her 1000 kişi içerisindeki girişimci sayısı bilgisini tamsayı formatında veren kayıttır.
15. A15: Şehirlere ait 1995 yılı suç işlenme oranlarının tamsayı formatında tutulduğu kayıttır.

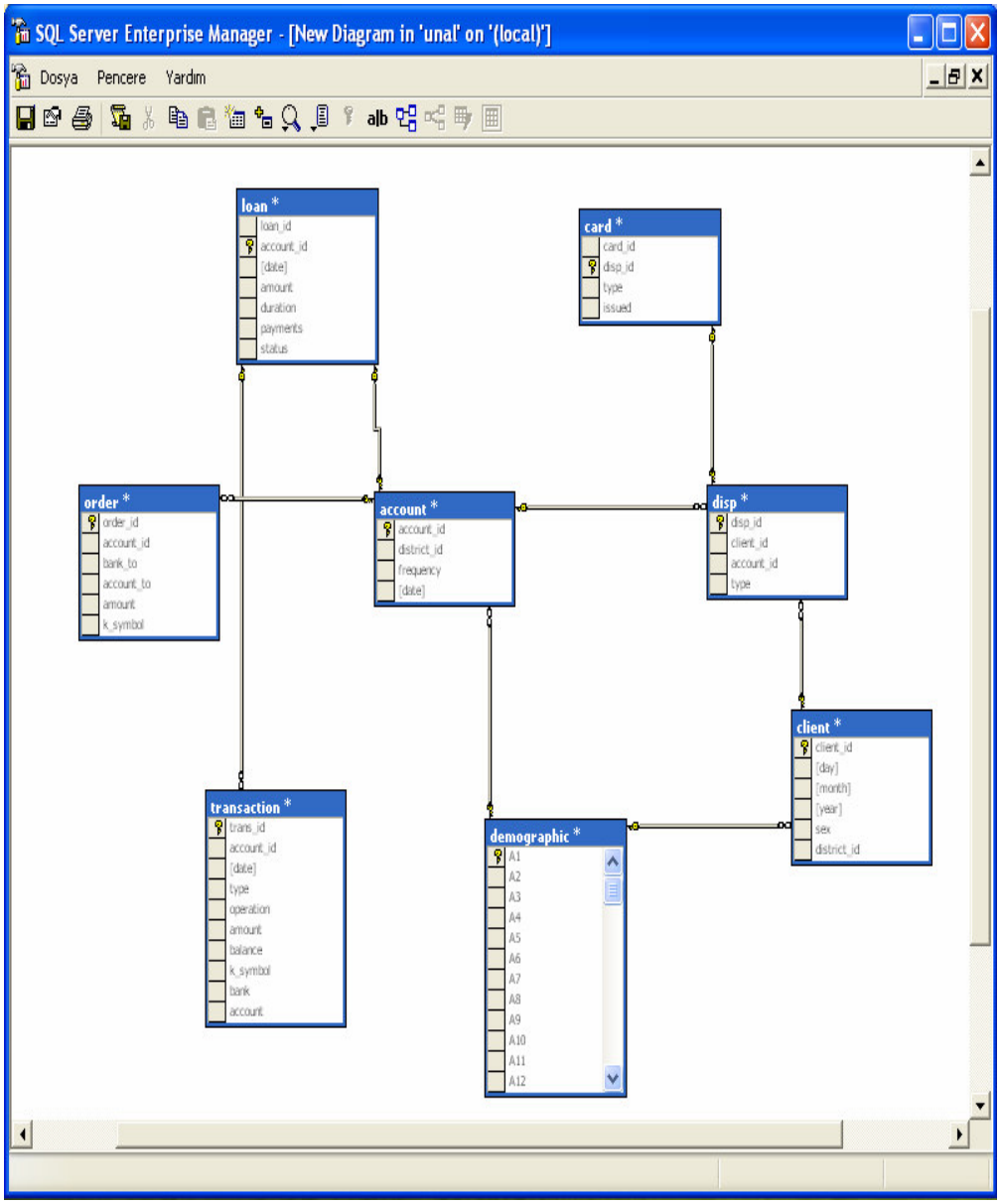
16: A16: Şehirlere ait 1996 yılı suç işlenme oranlarının tamsayı formatında tutulduğu kayıttır.

5.1.2 Veri Önişleme Süreci

Veri ambarlarında tutulan verilerin ele alınan problemin yapısına göre seçilmesi, hazırlanması ve analize imkan verecek şekilde yapılandırılması gerekmektedir (Özmen, 2003). Bu nedenle mevcut ham veri, bazı veri önişlemlerinden geçirilebilmektedir.

Veri önişleme, ham verinin veri madenciliği sürecindeki kullanımına hazır hale getirildiği aşamadır. Ham veri içerisinde tamamlanmamış (incomplite), tutarsız (inconsistent), gürültülü (noisy), ilişkisiz (irrelevant) vb. veriler bulunabilir. Bu nedenle veriyi önişlemden geçirmek, veri madenciliği sürecinin etkinliğini artırmakta ve aynı zamanda sürecin daha sorunsuz sürdürülmesini sağlamaktadır.

Veriler üzerindeki veri önişlemlerinin rahat bir şekilde gerçekleştirilmesi, tabloların birleştirilerek anlamsal bir bütünlük içerisinde tutulması gibi ihtiyaçlardan dolayı veriler, SOL Server 2000 veritabanı programına dahil edilmiştir (Şekil 5.1). 8 farklı tabloda yer alan verilerin, müşterilerin hesap numaraları referans alınarak 4500 kayıtlık bir alt tabloya indirgenmesi hedeflenmiştir. Öncelikle farklı tablolarda yer alan kayıtlar veri önişlemeden geçirilmiş ve bu tablolardan veri madenciliği sürecinde kullanılması hedeflenen kayıtlar; SQL (Structured Query Language) sorgulamaları ile çekilip, özet bir tabloda bir araya getirilmişlerdir.



Şekil 5.1 Verilerin SQL Server 2000'deki gösterimi.

5.1.2.1 Veri Temizleme Aşaması

Veri temizleme aşamasında, eksik verilerin tamamlanması, gürültü olarak nitelendirdiğimiz veri içerisindeki hataların ve tutarsız verilerin düzeltilmesi hedeflenmektedir.

8 farklı tabloda yer alan veriler üzerinde SQL sorguları ile eksik veya tutarsız kayıt olup olmadığı incelendi. Tablolarda yer alan kayıtlarda herhangi eksik veya tutarsız veriye rastlanmadı.

Elde edilen banka verisinde yer alan değişkenlerin isimlendirilmesi İngilizce ile yapılmıştır.

Fakat aynı durum kayıtların büyük bölümünün isimlendirilmesinde tekrar etmemiştir. Kayıtlarda genel olarak Çekçe isimlendirme yapılmış, bu durum ise verinin incelenmesi sürecinde sıkıntı oluşturmuştur. Verilerde yer alan kayıtların daha iyi anlaşılması amacıyla veride yer alan Çekçe kelimeler İngilizce karşılıkları ile değiştirilmiştir. Çizelge 5.1'de hangi tabloda hangi değişiklerin yapıldığı bilgisi yer almaktadır.

Çizelge 5.1 Tablolarda hangi değişikliklerin yapıldığı bilgisi

Çizelge Adı	**Değişkenin Eski Adı**	**Değişkenin Yeni Adı**
Account	POPLATEK MESICNE	Monthly
	POPLATEK TYDNE	Weekly
	POPLATEK PO OBRATU	LessThanWeekly
Permanent Order	POJISTNE	Insurance
	SIPO	Household
	UVER	Loan
Transaction	SLUZBY	Statement
	UROK	InterestCredit
	SANKC. UROK	NegativeBalance
	DUCHOD	Pension

Verinin daha anlaşılır olması amaçlanarak yapılan önişlemlerden bir diğeri ise Client tablosunda yer alan "birth_number" sütunu üzerinde gerçekleştirilmiştir. Müşterinin doğum tarihi ile cinsiyeti bilgisi "birth_number" sütununda; bayanlar için YYMM+50DD, erkekler için ise YYMMDD formatında tek bir sütunda tutulmaktadır (Şekil 5.2).

	client_id	birth_number	district_id
1	1	706213	18
2	2	450204	1
3	3	406009	1
4	4	561201	5
5	5	605703	5
6	6	190922	12
7	7	290125	15
8	8	385221	51

Şekil 5.2 Yaş ve cinsiyet bilgisinin aynı sütunda tutulduğu durum

SQL sorgulamaları ile "birth_number" sütunu, *birthdate* ve *sex* sütunlarına ayrıştırıldı. Böylece hem verinin okunabilirliği artırılmış hem de veri madenciliği sürecinde verinin daha kolay işlenebilmesi sağlanmış oldu (Şekil 5.3).

	client_id	birthdate	sex	district_id
1	1	1970-12-13 00:00:00	F	18
2	2	1945-02-04 00:00:00	M	1
3	3	1940-10-09 00:00:00	F	1
4	4	1956-12-01 00:00:00	M	5
5	5	1960-07-03 00:00:00	F	5
6	6	1919-09-22 00:00:00	M	12
7	7	1929-01-25 00:00:00	M	15
8	8	1938-02-21 00:00:00	F	51

Şekil 5.3 Yaş ve cinsiyet bilgisinin ayrı sütunlarda tutulduğu durum

5.1.2.2 Veri Türetme ve Veri Seçilmesi Aşaması

Verinin seçilmesi; veri madenciliği sürecinde kullanılmaya uygun olan verinin seçilmesi aşamasıdır. Bu aşama, analizde kullanılacak değişkenlerin ve bu değişkenleri oluşturan kayıtların seçilmesi olarak ikiye ayrılabilir.

Bu aşamada, 8 farklı tabloda bulunan ham veri, müşterilerin hesap numaraları referans alınarak 4500 kayıtlık bir özet tabloya indirgenmiştir. Bu özet tablonun oluşturulması sırasında, veri madenciliği sürecinde kullanılmayacak olan bazı kayıtlar veriden silinmiş, veri madenciliği sürecinde ele alınacak olan problemin çözümüne katkı sağlayabilecek yeni kayıtlar ise mevcut veriden türetilmişlerdir.

Tablolarda bulunan kayıtları numaralandıran ve diğer tablolar ile ilişki kurulmasına yardımcı

olan “account_id”, “trans_id” vb. kayıtların, veri madenciliği sürecine herhangi bir katkısı olmayacağından dolayı, özet tabloda yer verilmemiştir (Çizelge 5.2).

Ayrıca, Transaction tablosunda yer alan “operation” kayıdı ile yine aynı tabloda yer alan “k_symbol” kayıdının her ikisi de müşterinin hesap üzerindeki işlem hareketi hakkında benzer bilgileri tutmaktadırlar. Benzer yapıya sahip olan bu iki kayıttan “operation” kayıdına özet tabloda yer verilmemiştir (Çizelge 5.2).

Demographic tablosu, müşterilerin ve banka şubelerinin bulunduğu bölgelerin bağlı olduğu şehirler ile ilgili ekonomik ve sosyal bilgi vermektedir. Bu nedenle sadece müşterinin yaşadığı şehir ile ilgili bilgi veren A3 kaydına özet tabloda yer verilmiştir (Çizelge 5.2).

Çizelge 5.2 Ham veride yer alıp da özet tabloda yer almayan kayıtlar.

Çıkarılan Kayıtlar	**Çıkarılan Kayıdın Bulunduğu Tablo**
Account_id, district_id	Account
Client_id, district_id	Client
Disp_id, account_id	Disposition
Order_id, account_id, account_to, bank_to	Permanent Order
Trans_id, account_id, operation, bank, account	Transaction
Loan_id, account_id	Loan
Card_id, disp_id	Credit card
A1, A2, A4, A5, A6, A7, A8, A9, A10, A11, A12, A13, A14, A15, A16	Demographic

Mevcut veriden türetilen yeni değişkenler Çizelge 5.3 ve Çizelge 5.4‘de gösterilmektedir.

Çizelge 5.3 Özet tabloda yer alan Nümerik-Sürekli (Continuous) değişkenler.

Değişken Adı	**Değişkenin Tanımı**	**Veri Tipi**	**Değişken Türetildi mi?**
ClientAge	Müşterinin yaşı	Tamsayı	Evet
LoanAmount	Müşterinin aldığı kredinin miktarı.	Tamsayı	Hayır
LoanDuration	Müşterinin, krediyi geri ödeme süresi (ay olarak)	Tamsayı	Hayır
LoanPayments	Müşterinin krediyi geri ödemesi sırasında aylık yatırması gereken para miktarı	Tamsayı	Hayır
OrderAvgAmount	Müşterinin yapmış olduğu otomatik ödeme işlemlerinin miktarının ortalaması	Tamsayı	Evet
CountInsurance	Müşterinin kaç defa hesabından sigorta ödemesi yaptığı bilgisi	Tamsayı	Evet
AvgInsurance	Yapılan sigorta ödemeleri miktarı ortalaması	Tamsayı	Evet
CountStatement	Alınan hesap özetlerinin sayısı	Tamsayı	Evet
AvgStatement	Alınan hesap özetleri için ortalama ödenen miktar	Tamsayı	Evet
CountInterestCredit	Müşteri hesabına yapılan faiz ödemelerinin sayısı	Tamsayı	Evet
AvgInterestCredit	Müşteri hesabına yatırılan ortalama faiz miktarı	Tamsayı	Evet

Değişken Adı	Değişkenin Tanımı	Veri Tipi	Değişken Türetildi mi?
CountHousehold	Müşterinin kaç defa hesabından ev giderleri (su, telefon vb.) ödemesi yaptığı bilgisi	Tamsayı	Evet
AvgHousehold	Müşterinin ortalama yaptığı ev giderleri harcaması	Tamsayı	Evet
CountOldagePension	Müşterinin hesabına kaç defa emeklilik ödemesi yapıldığı bilgisi	Tamsayı	Evet
AvgOldagePension	Müşterinin hesabına yapılan emeklilik ödemesinin ortalama miktarı	Tamsayı	Evet
CountLoanPay	Müşterinin kaç defa hesabından kredi ödemesi yaptığı bilgisi	Tamsayı	Evet
Balance	Müşterinin yaptığı işlemler sonrası hesabında kalan para miktarı	Tamsayı	Hayır
AvgTransIn	Müşterinin hesabına ortalama giren para miktarı	Tamsayı	Evet
AvgTransOut	Müşteri hesabından çıkan ortalama para miktarı	Tamsayı	Evet
LengthOfCustomer	Kaç aydan beri banka müşterisi olduğu bilgisi	Tamsayı	Evet
ClientAvgFreq	Yapılmış olan hesap işlemleri sayısının ortalaması	Tamsayı	Evet

Çizelge 5.4 Özet tabloda yer alan Nominal-Kategorik (Categorical) Değişkenler

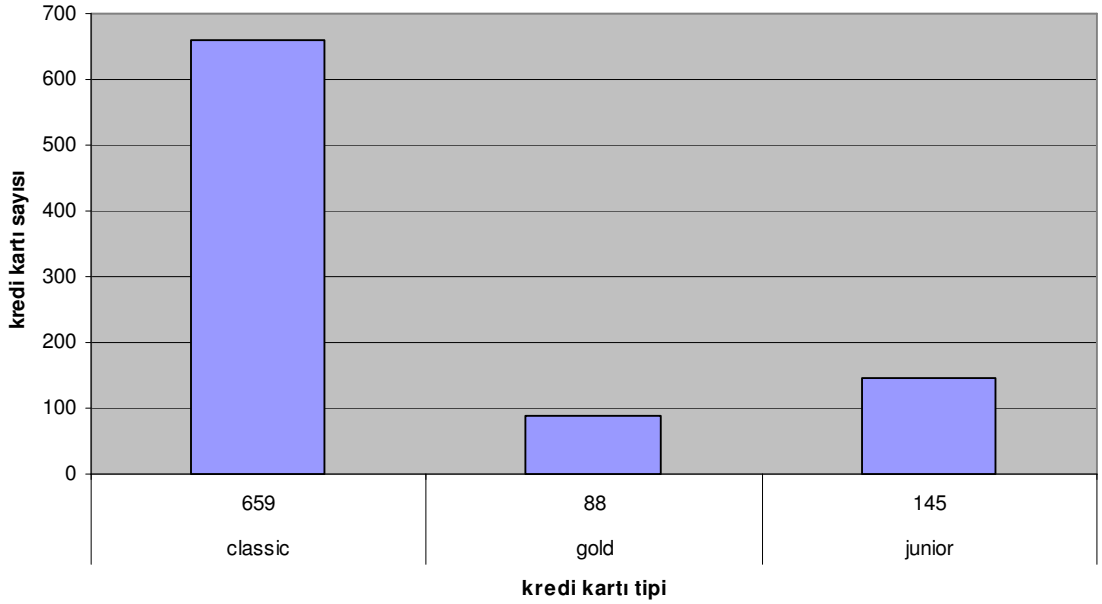
Değişken Adı	**Değişkenin Tanımı**	**Veri Üzerinde Yapılan Dönüştürme**	**Veri Tipi**	**Değişken Türetildi mi?**
ClientAccFreq	Müşterinin hesabını kullanma sıklığı	0: Monthly 1: Weekly 2: LessThanWeekly	Tamsayı	Hayır
ClientRegion	Hesabın açıldığı şehir	1: Prague 2: South Moravia 3: North Moravia 4: South Bohemia 5: North Bohemia 6: East Bohemia 7: West Bohemia 8: Central Bohemia	Tamsayı	Hayır
ClientSex	Müşterinin cinsiyeti	0: Male (Erkek) 1: Female (Bayan)	Tamsayı	Evet
AccDispStatus	Hesabı kullanan kişinin hesap üzerindeki yetkisi	0: Başka bir kullanıcı için hesap üzerinde işlem yapma izni verilmiş 1: Başka bir kullanıcı için hesap üzerinde işlem yapma izni verilmemiş	Tamsayı	Evet
LoanStatus	Müşterinin aldığı kredinin statüsü (A, B, C, D)	Kredi Kartı Tipi veri seti için bağımlı değişken	Metin	Hayır
CreditCardType	Müşterinin sahip olduğu kart tipi (classic,junior,gold)	Kredi Statüsü veri seti için bağımlı değişken.	Metin	Hayır

Veri setinin önişlemden geçirilmesi sırasında, dikkati çeken iki çıkarsama yapılmıştır:

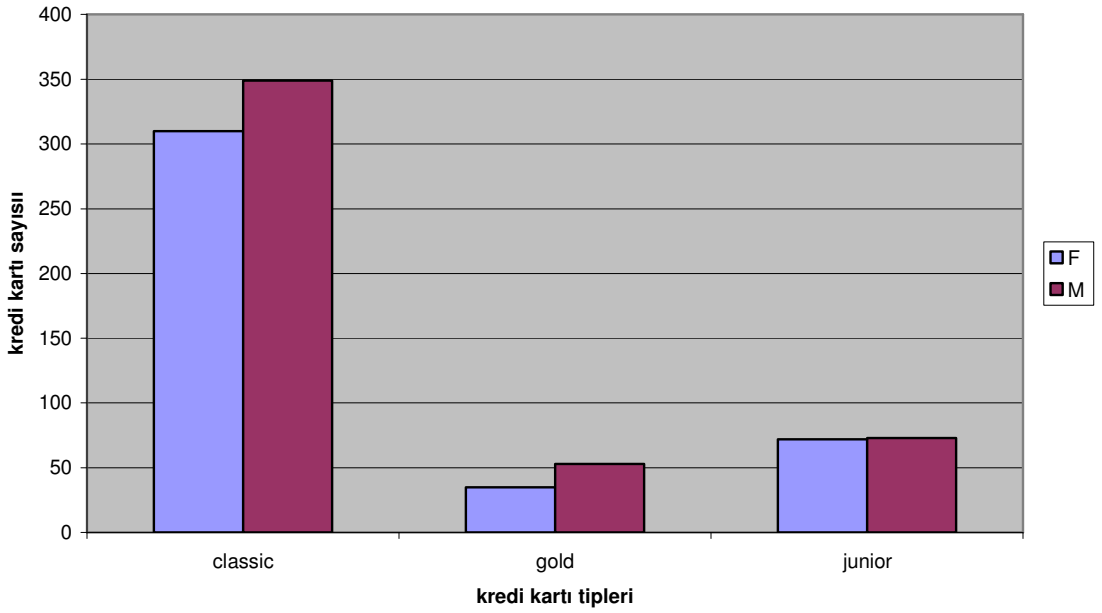
- Bir hesap üzerinde bir veya iki müşteri işlem yapabilir. Ama bir müşteri sadece tek hesap üzerinde işlem yapabilir.
- Bankadan bir başka bankaya yapılan işlemlerde, paranın transfer edildiği bankanın adı ve o bankadaki hesap numarası bilgisi veri setinde yer almaktadır. Bu kayıtlar dikkatle incelendiğinde paranın genellikle iki hesap arasında transferinin gerçekleştirildiği görülmüştür.

Ham verinin, veri madenciliği sürecinde kullanılmak üzere bazı veri önişlemlerinden geçirilerek indirgenmesi ile özet tablo elde edilmiştir. Bu özet tablo üzerinde gerçekleştirilen SQL sorguları ile 682 kayıtlı Kredi Statüsü veri seti ve 892 kayıtlı Kredi Kartı Tipi veri seti oluşturulmuştur. Bu iki veri setinde aynı bağımsız değişkenler yer almaktadır (Çizelge 5.3 ve Çizelge 5.4'de yer alan değişkenler). Sadece üzerinde sınıflandırma ve kümeleme analizlerinin gerçekleştirileceği bağımlı değişkenler farklıdır. Kredi Kartı Tipi veri setinde bağımlı değişken olarak Çizelge 5.4'de gösterilen CreditCardType değişkeni, Kredi Statüsü veri setinde ise bağımlı değişken olarak yine Çizelge 5.4'de gösterilen LoanStatus değişkeni kullanılacaktır.

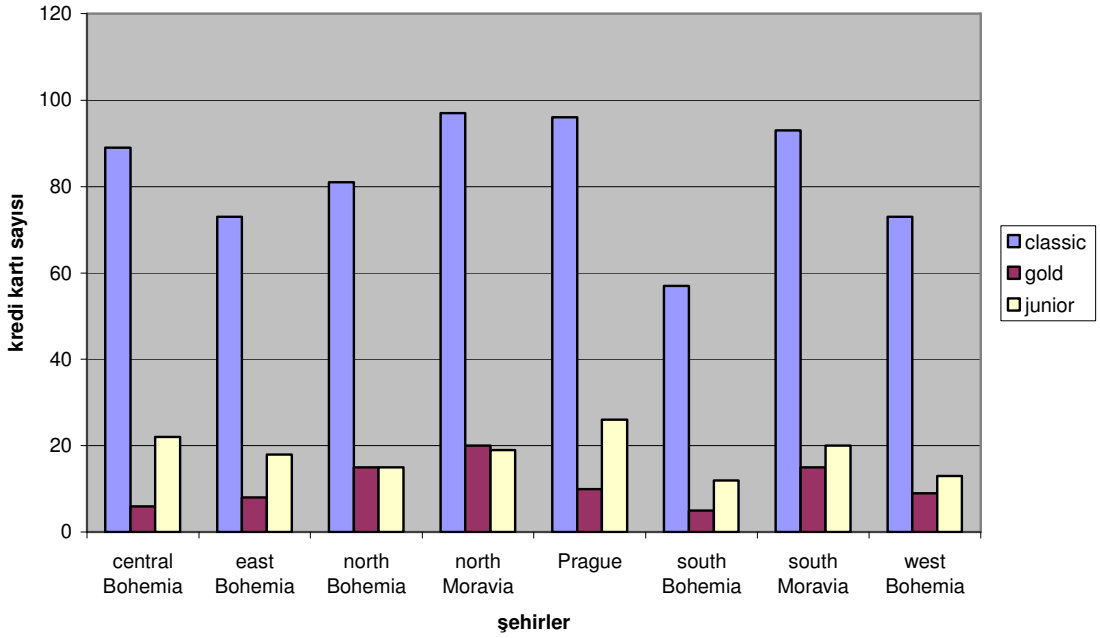
Kredi Kartı Tipi veri setinde yer alan kredi kartları içerisinde Classic kart, sayısal olarak en fazla bulunan kart türüdür (Şekil 5.4). Kredi kartlarının bayan ve erkek müşteriler arasındaki dağılımı incelendiğinde, Classic ve Gold kart sahibi müşterilerde erkek müşterilerin bayanlara göre sayısal olarak biraz daha fazla olduğu, Junior kartta ise sayısal olarak bayan ve erkek dağılımının hemen hemen aynı olduğu görülmektedir (Şekil 5.5). Classic kart ve Gold karta en çok North Moravia şehrinde yaşayan müşteriler sahip iken, Junior karta en çok Prague şehrinde oturan müşteriler sahiptir (Şekil 5.6).



Şekil 5.4 Kredi kartı türlerinin sayılarına göre dağılım grafiği.

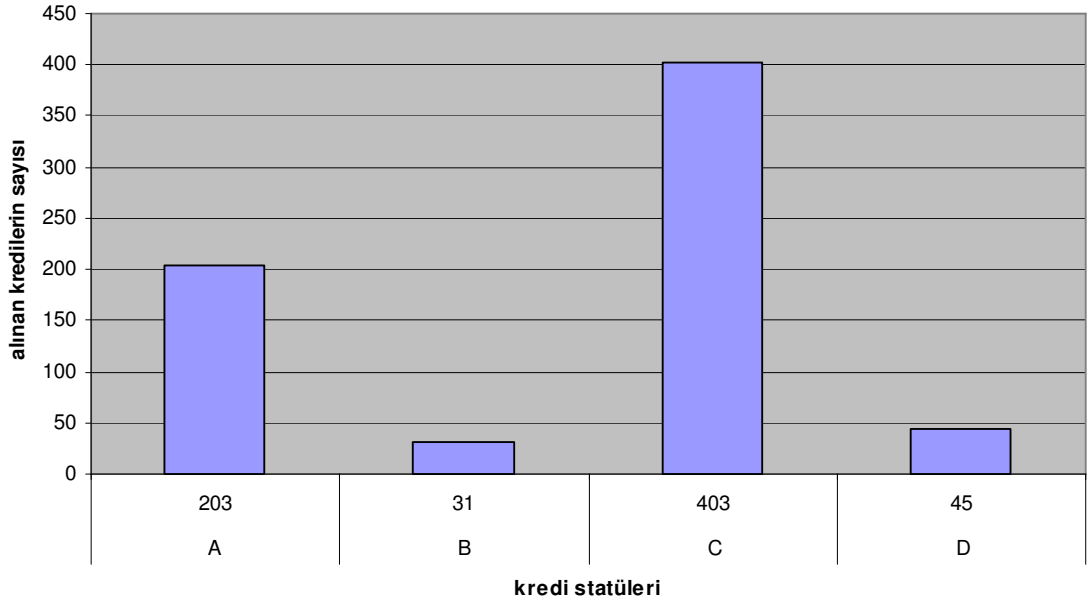


Şekil 5.5 Kredi kartı türlerinin müşterinin cinsiyetine göre dağılım grafiği.

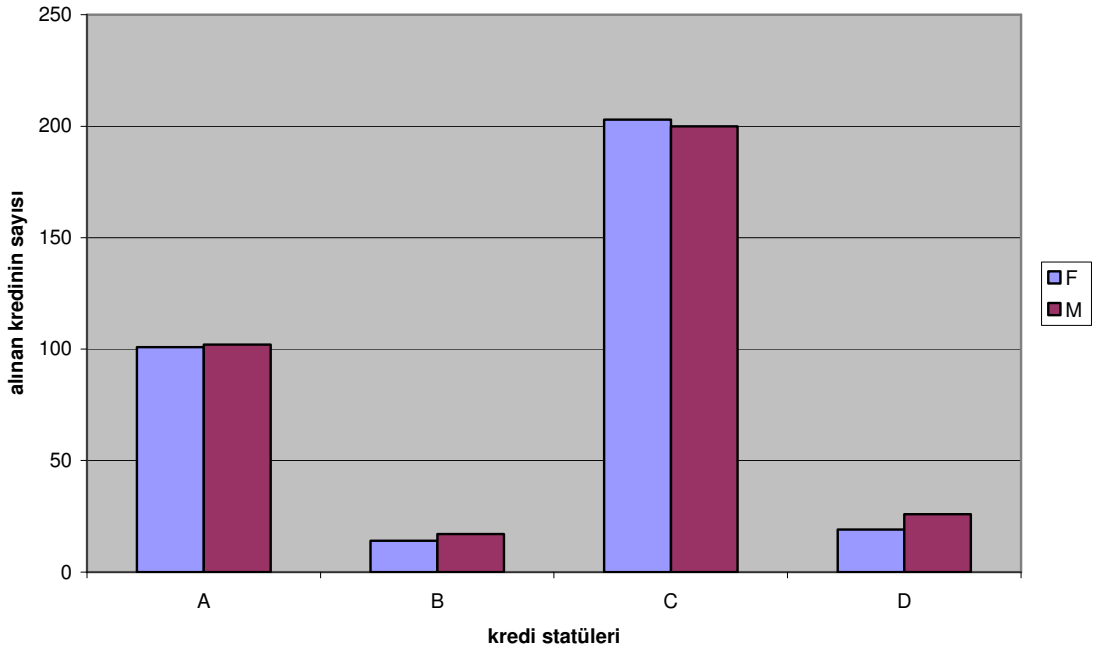


Şekil 5.6: Kredi kartlarının, müşterilerin yaşadıkları şehirlere göre dağılım grafiği.

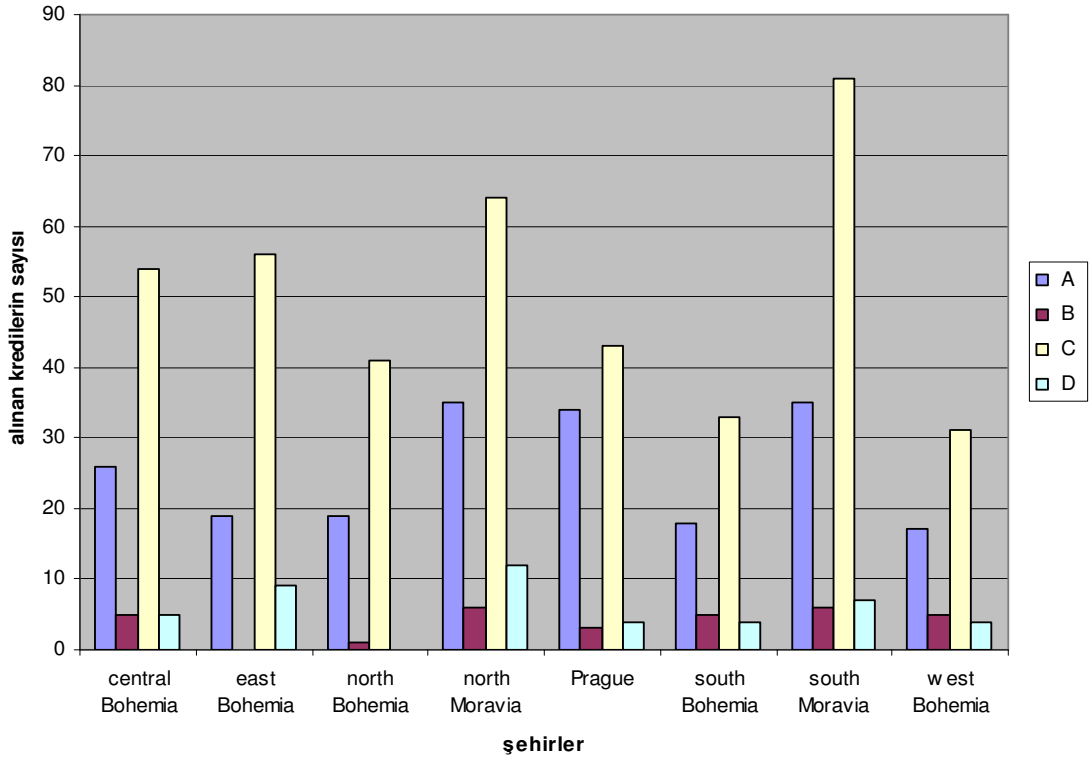
Kredi Statüsü veri setinde yer alan kredilerin sayısal dağılımı incelendiğinde, müşterilerin kredi statülerinin ağırlıklı olarak A veya C olduğu görülmektedir (Şekil 5.7). A, B, C ve D kredi statülerinin bayan ve erkek müşteriler arasındaki dağılımı incelendiğinde; A, B ve D kredi statülerinde az farkla erkeklerin sayısal olarak fazla olduğu, C kredi statüsünde ise sayısal olarak az farkla bayanların fazla olduğu görülmektedir (Şekil 5.8). A, B ve C kredi statüsüne sahip müşteriler en çok North Moravia şehrinde otururken, D kredi statüsüne sahip müşteriler en çok South Moravia şehrinde oturmaktadır (Şekil 5.9).



Şekil 5.7 Alınan kredinin statüsünün, sayılarına göre dağılım grafiği.



Şekil 5.8 Alınan kredinin statüsünün, krediyi alan müşterinin cinsiyetine göre dağılımı.



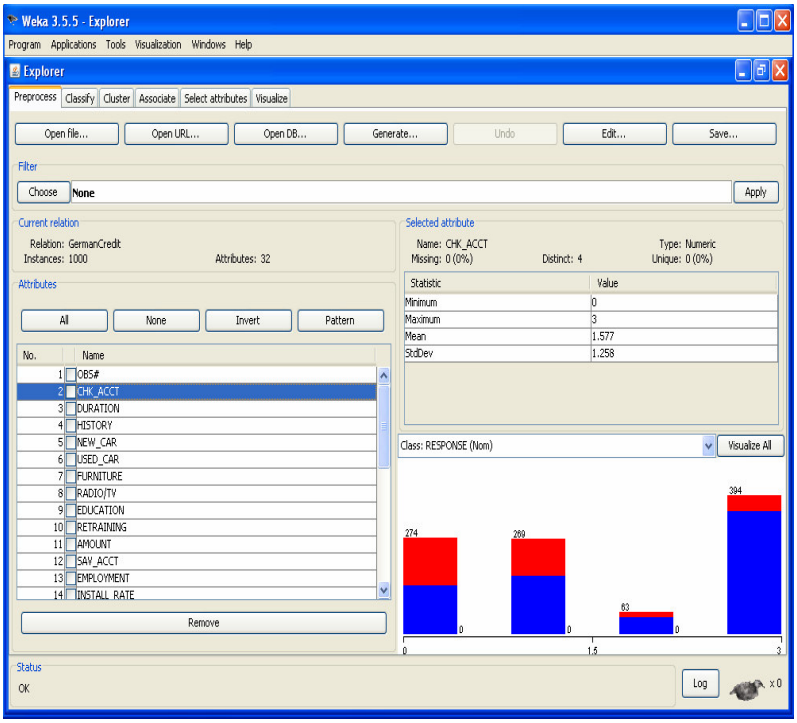
Şekil 5.9 Alınan kredinin statüsünün, müşterilerin yaşadıkları şehirlere göre dağılım grafiği.

5.2 Kredi Kartı Tipi ve Kredi Statüsü Veri Setlerinin Veri Analizinin Gerçekleştirilmesi

Uygulamanın bundan sonraki sürecinde, farklı sınıflandırma algoritmalarının Kredi Kartı Tipi ve Kredi Statüsü veri setleri üzerindeki sınıflandırma başarıları test edilecek ve ayrıca Kredi Kartı Tipi veri seti için bir model tanımlanacaktır.

Veri setlerinin sınıflandırılmasında, Java programlama dili ile yazılmış olan Weka* (Waikato Environment for Knowledge Analysis) veri madenciliği paket programı kullanılacaktır. Weka veri madenciliği paket programı; veri dosyalarının yüklendiği ve veri üzerinde bazı önişlemlerin yapılabildiği Preprocess (Önişleme) paneli, birçok sınıflandırma algoritmasının bulunduğu Classifier (Sınıflandırıcı) paneli, veri madenciliğinde sık kullanılan bazı kümeleme algoritmalarının yer aldığı Cluster (Kümeleme) paneli, veri kümesi üzerinde birliktelik kurallarının uygulandığı Associate (Birliktelik) paneli ve farklı nitelikler (attributes) arasındaki ilişkileri grafiksel olarak gösteren Visualize (Görselleştirme) panelinden oluşmaktadır. Weka'nın ekran görünümü Şekil 5.10'da yer almaktadır.

* Weka paket programı, http://www.cs.waikato.ac.nz/ml/weka adresinden temin edilebilir.



Şekil 5.10 Weka paket programı ekran görüntüsü

Uygulamada kullanılacak olan veri setlerinin Weka'ya yüklenebilmesi için, Weka programı tarafından tanınan bir formata dönüştürülmeleri gerekmektedir. Bu nedenle Kredi Kartı Tipi ve Kredi Statüsü veri setleri, Excel çalışma kitabında yer alan CSV (Comma Seperated Values) formatına dönüştürülerek Weka'ya yüklenmiştir. Weka programı sınıflandırma veya kümeleme işlemleri çıktılarını ARFF (Attribute Related File Format) formatında oluşturmaktadır. CSV formatında yüklenen veri setleri istenirse, *Önişleme panelinde* bulunan *Save düğmesi* ile ARFF formatına dönüştürülerek saklanabilir.

Weka veri madenciliği programında kullanılan algoritmaların ürettiği çıktılarda, yapılan sınıflandırma sonuçlarının anlamlılığı ile ilgili bilgi veren bazı parametreler vardır. Veri setlerinin analizi sırasında kullanacağımız J48, Çok Katmanlı Algılayıcı (ÇKA), Lojistik Regresyon (LR) algoritmalarının hepsinde ortak olarak kullanılan parametreler şunlardır:

- Confusion Matrix: Algoritmalarda yer alan sınıf etiketi değerlerinin hangi sınıflara atandığı ile ilgili bilgi vermektedir. Doğru olarak sınıflandırılan kayıtların sayısı, matrisin diyagonal elemanlarının toplamına eşittir.
- Başarı Oranı (Correctly Classified Instances): Doğru olarak sınıflandırılan kayıtların yüzde oranı hakkında bilgi verir.
- Hata Oranı (Incorrectly Classified Instances): Yanlış olarak sınıflandırılan kayıtların

yüzde oranı hakkında bilgi verir.

- Kappa İstatistik (Kappa Statistic): İki veya daha fazla gözlem arasındaki kalitatif uyum oranını ölçmek için geliştirilmiş bir testtir. Kappa değeri 1'e yakınsadığında gözlemler arasındaki uyumun arttığını, 0' a yakınsadığında ise uyumun azaldığını söyleyebiliriz.

5.2.1 Kredi Kartı Tipi Veri Setinin Sınıflandırılması

Uygulamanın bu aşamasında J48, ÇKA ve LR algoritmalarının Kredi Kartı Tipi veri seti üzerindeki sınıflandırma başarıları test edilecektir. CreditCardType (classic, junior, gold) kayıdı, hedef değişken veya sınıf etiketi olarak belirlenmiştir.

5.2.1.1 Kredi Kartı Tipi Veri Setinin J48, ÇKA ve LR Algoritmaları ile Sınıflandırılmasının Gerçekleştirilmesi

J48 algoritması; sınıflandırma algoritması olarak geniş bir kullanım alanına sahip olan ID3 (C4.5) algoritmasının Weka veri madenciliği programında kullanılmasına imkan sağlayacak şekilde düzenlenmesi elde edilen bir algoritmadır. J48 algoritması, ID3 algoritmasından farklı olarak nümerik değerler üzerinde sınıflandırma yapabilir. Bunun yanında, ID3 algoritmasında karar ağacı oluşturulduktan sonra yapılan budama işlemi, J48 algoritması ile karar ağacının oluşturulması esnasında gerçekleştirilir.

J48, ÇKA ve LR algoritmaları tarafından kullanılacak olan eğitim ve test verilerinin ayrılması, sınıflandırma panelinde yer alan Çapraz Geçerleme (Cross Validation) metodunun seçilmesi ile gerçekleştirilmiştir. Çapraz Geçerleme yönteminde veri kümesi m eş büyüklükte alt kümeye ayrılır (program default olarak m değerini 10 olarak almıştır). Her iterasyonda m-1 parça modelin öğretilmesinde, 1 parça ise modelin test edilmesinde kullanılmaktadır. Her bir iterasyon sonucu elde edilen hata değerlerinin aritmetik ortalaması alınarak, modelin başarı oranı ile ilgili bilgilere ulaşılır.

J48, ÇKA ve LR algoritmalarının Kredi Kartı Tipi veri setine uygulanması ile elde edilen çıkış değerleri Çizelge 5.5'de gösterilmektedir.

Çizelge 5.5 Kredi Kartı Tipi veri setine uygulanan sınıflandırma algoritmaları sonuçları.

Sınıflandırma Algoritmaları	**Confusion Matrix**				**Başarı Oranı (%)**	**Kappa İstatistik Değerleri**
J48	a	b	c		80,8296	0,516
	21	64	3	a=gold		
	39	591	29	b=classic		
	1	35	109	c=junior		
ÇKA	a	b	c		77,4664	0,4273
	19	65	4	a=gold		
	50	579	30	b=classic		
	1	51	93	c=junior		
LR	a	b	c		80,9417	0,4789
	9	74	5	a=gold		
	15	614	30	b=classic		
	1	45	99	c=junior		

Kredi Kartı Tipi veri setine uygulanan sınıflandırma algoritmalarının ürettiği sonuçlar incelendiğinde, en başarılı sınıflandırmanın LR algoritması tarafından gerçekleştirildiği görülmektedir.

5.2.1.2 Nitelik Seçme Panelinin Kredi Kartı Tipi Veri Setine Uygulanması

Weka'da bulunan *Nitelik Seçme (Select Attribute) paneli*, veride yer alan bağımsız değişkenlerin bağımlı değişken (sınıf etiketi) üzerindeki açıklayıcılıkları ile ilgili bilgi vermektedir. Hangi bağımsız değişkenlerin bağımlı değişken üzerinde ne kadar açıklayıcı olduğunu belirlemek için Kredi Kartı Tipi veri seti, *Nitelik Seçme panelinde* işleme sokulmuştur. Elde edilen sonuçlar Çizelge 5.6'da gösterilmektedir.

Çizelge 5.6 Veri setinde yer alan bağımsız değişkenlerin, bağımlı değişken üzerindeki açıklayıcılıkları

Attribute Evaluator (supervised, Class (nominal): CreditCardType):
Information Gain Ranking Filter

Ranked attributes:		Ranked attributes:	
0,3671	ClientAge	0	ClientRegion
0,0969	Balance	0	AvgHousehold
0,0385	AvgTransIn	0	AvgInterestCredit
0,033	OrderAvgAmount	0	CountInterestCredit
0,0218	AvgTransOut	0	CountInsurance
0,0173	CountHousehold	0	AvgInsurance
0	LengthOfCustomer	0	CountStatement
0	ClientSex	0	CountOldagePension
0	AvgStatement	0	CountLoanPay
0	LoanAmount	0	AccDispStatus
0	AvgOldagePension	0	LoanDuration
0	ClientAvgFreq	0	LoanPayments
0	ClientAccFreq		

Çizelge 5.6'dan da açıkça görüldüğü gibi; kredi kartı için başvuran müşterinin yaş bilgisini tutan *ClientAge*, müşterinin hesabında yaptığı işlemler sonrası kalan parası ile ilgili bilgi tutan *Balance*, müşterinin hesabından çıkan veya hesabına giren paranın ortalama miktar bilgisini tutan *AvgTransIn* ve *AvgTransOut*, müşterinin hesabından yapmış olduğu otomatik ödemelerin miktarı bilgisini tutan *OrderAvgAmount* ve müşterinin hesabından yapmış olduğu ev giderleri (su, elektrik vb.) harcamalarının sayısı ile ilgili bilgi veren *CountHousehold* bağımsız değişkenleri, bağımlı değişken üzerinde açıklayıcılığa sahip değişkenlerdir. Elde edilen bu 6 bağımsız değişken, J48, ÇKA ve LR algoritmaları tarafından yeniden sınıflandırılmışlardır. Sınıflandırmalar sonucu elde edilen değerler Çizelge 5.7'de gösterilmektedir.

Çizelge 5.7 Nitelik seçme paneli ile indirgenen veriye uygulanan sınıflandırma algoritmaları sonuçları

Sınıflandırma Algoritmaları	**Confusion Matrix**				**Başarı Oranı (%)**	**Kappa İstatistik Değerleri**
J48	a	b	c		82,287	0,5157
	8	79	1	a=gold		
	16	618	25	b=classic		
	1	36	108	c=junior		
ÇKA	a	b	c		82,1749	0,5212
	14	72	2	a=gold		
	22	613	24	b=classic		
	0	39	106	c=junior		
LR	a	b	c		81,6143	0,4851
	8	79	1	a=gold		
	14	621	24	b=classic		
	0	46	99	c=junior		

Nitelik seçme paneli ile indirgenen veriye uygulanan sınıflandırma algoritmalarının ürettiği başarı oranı değerlerinin (Çizelge 5.7), özet veriye uygulanan sınıflandırma algoritmalarının ürettiği başarı oranı değerlerine (Çizelge 5.5) göre daha iyi sonuçlar verdiği görülmektedir. Algoritmaların başarı oranları incelendiğinde; J48 algoritmasında yaklaşık %1,5, ÇKA'da yaklaşık %5, LR'da ise yaklaşık %1'lik artış elde edildiği görülmektedir. Başarı oranlarındaki bu artışın, Kappa istatistik değerlerindeki artışa da yansıdığı görülmektedir.

5.2.1.3 Kredi Kartı Tipi Veri Setine Uygulanan J48 ve ÇKA Sınıflandırma Algoritmalarının Parametrelerinin Değiştirilmesi

Farklı değerlerinin sınıflandırmadaki başarı oranını etkilemesi muhtemel J48 algoritması parametreleri şunlardır:

- Güven Faktörü (Confidence factor): Bu parametre, karar ağacının dengeli büyümesine yardımcı olan budama işleminin etkinliğini artırmak için kullanılmaktadır. Bu parametrenin alacağı küçük değerler, daha fazla budamanın gerçekleştirilmesini sağlamaktadır.
- MinNumObj: Her yaprakta yer alması gereken kayıt sayısı bilgisini ifade etmektedir.

J48 algoritmasında yer alan Güven Faktörü ve MinNumObj parametrelerine atanan varsayılan değerler değiştirilerek sınıflandırmadaki başarı oranı artırılmaya çalışılmıştır. J48 algoritmasının değişik Güven Faktörü değerlerine karşılık gelen en yüksek 5 başarı oranı değeri ve bu değerlere karşılık gelen Kappa istatistik sonuçları Çizelge 5.8'de gösterilmektedir.

Çizelge 5.8 Farklı Güven Faktörü değerlerine karşılık gelen Başarı Oranı ve Kappa İstatistik sonuçları

Güven Faktörü	**Başarı Oranı (%)**	**Kappa İstatistik**
0,15	83,296	0,5275
0,14	83,296	0,5236
0,13	83,296	0,5236
0,08	83,5202	0,5214
0,07	83,5202	0,5214

Güven Faktörü 0,15 değeri için üretilen sonuçlar incelendiğinde (Çizelge 5.8), J48 algoritmasının sınıflandırma başarısının (Çizelge 5.7) yaklaşık %1 oranında artmış olduğu görülmektedir.

Çizelge 5.8'de de görüldüğü üzere; 0.08 ve 0.07 Güven Faktörü değerlerinin başarı oranları, 0,15 Güven Faktörü değerinin ürettiği başarı oranından yüksek olmasına rağmen, gerek modelin anlamlılığının bir ölçütü olan Kappa istatistik değerlerinin 0,15 Güven Faktörü değerinden küçük olması, gerekse de bu değerlere karşılık gelen ağaç yapısında budama işleminin fazla yapılması (bu nedenle yorumlanabilirlikte sıkıntı oluşmaktadır) nedeniyle Güven Faktörü olarak 0,15 değeri, uygulamanın sonraki adımı için kullanılmaya uygun görülmüştür.

Güven Faktörü değeri 0,15 olarak seçilen J48 algoritması, farklı MinNumObj değerlerine karşılık gelen yeni Başarı Oranı ve Kappa İstatistik sonuçları üretmiştir. Farklı MinNumObj değerleri için elde edilen en yüksek 5 Başarı Oranı değeri ve bu değerlere karşılık gelen Kappa İstatistik sonuçları Çizelge 5.9'da gösterilmektedir.

Çizelge 5.9 Farklı MinNumObj değerlerine karşılık gelen Başarı Oranı ve Kappa İstatistik sonuçları

MinNumObj (Güven Faktörü=0,15)	**Başarı Oranı (%)**	**Kappa İstatistik**
7	83,296	0,5304
5	83,296	0,5283
6	83,296	0,5244
4	82,8475	0,5234
12	83,0717	0,5179

Çizelge 5.8 ve Çizelge 5.9'da yer alan sonuçlar karşılaştırıldığında; farklı MinNumObj değerlerinin, J48 algoritmasının sınıflandırmadaki başarı oranı üzerinde bir artışa neden olmadığı, Kappa İstatistik değerinde ise çok az bir artışa (yaklaşık 0,003) neden olduğu görülmüştür. J48 algoritmasının Güven Faktörü ve MinNumObj parametrelerinin farklı değerlerinin test edilmesi sonucunda, sınıflandırmadaki başarı oranı %83,4081 ve bu orana karşılık gelen Kappa istatistik değeri ise 0,5589 olarak bulunmuştur.

Çizelge 5.5 ile Çizelge 5.9'da yer alan J48 algoritması sınıflandırma sonuçları incelendiğinde; J48 algoritmasının sınıflandırmadaki başarı oranında yaklaşık %3'lük, başarı oranına karşılık gelen Kappa İstatistik değerinde ise yaklaşık %2'lik bir artış sağlandığı görülmektedir.

ÇKA algoritmasında yer alan ve farklı değerlerinin, sınıflandırmadaki başarı oranını etkilemesi muhtemel 2 parametre vardır:

- Öğrenme Katsayısı (Learning Rate): Ağırlıkların değişim miktarını belirlemek için kullanılan bir katsayıdır. [0-1] aralığında keyfi bir değer alabilir.
- Momentum: Yerel çözümlere takılan ağların, bir sıçrama ile daha ileri sonuçlar üretmesini sağlayabilmek için önerilmiş bir katsayıdır. Momentum parametresi için [0-1] aralığında keyfi bir değer seçilir.

ÇKA algoritmasında yer alan öğrenme katsayısı ve momentum parametrelerine atanan varsayılan değerler değiştirilerek sınıflandırmadaki başarı oranı artırılmaya çalışılmıştır. ÇKA algoritmasının değişik öğrenme katsayısı değerlerine karşılık gelen en yüksek 5 Başarı Oranı değeri ve bu değerlere karşılık gelen Kappa İstatistik sonuçları Çizelge 5.10'da gösterilmektedir.

Çizelge 5.10 Farklı Öğrenme Katsayılarına karşılık gelen Başarı Oranı ve Kappa İstatistik sonuçları.

Öğrenme Katsayısı	**Başarı Oranı (%)**	**Kappa İstatistik**
0,5	83,296	0,5617
0,49	83,296	0,5566
0,35	82,9596	0,5498
0,34	82,7354	0,5457
0,33	82,6233	0,5401

Öğrenme katsayısının 0,5 değeri için elde edilen sonuçlar incelendiğinde (Çizelge 5.10), ÇKA algoritmasının sınıflandırmadaki başarı oranında (Çizelge 5.7) %1'den fazla, bu orana karşılık gelen Kappa İstatistik değerinde ise %4'den fazla bir artış sağlandığı gözlenmiştir.

ÇKA algoritması üzerinde etkin olabilecek bir başka parametre olan momentum parametresinin farklı değerlerinin Öğrenme Katsayısı 0,5 olarak düzenlendikten sonra algoritmada kullanılması ile elde edilen değerler Çizelge 5.11'de gösterilmektedir.

Çizelge 5.11 Farklı Momentum değerlerine karşılık gelen Başarı Oranı ve Kappa İstatistik sonuçları.

Momentum(Öğrenme Katsayısı=0,5)	**Başarı Oranı (%)**	**Kappa İstatistik**
0,19	83,4081	0,5589
0,2	83,296	0,5617
0,7	83,1839	0,5433
0,73	82,8475	0,5355
0,67	82,7354	0,5318

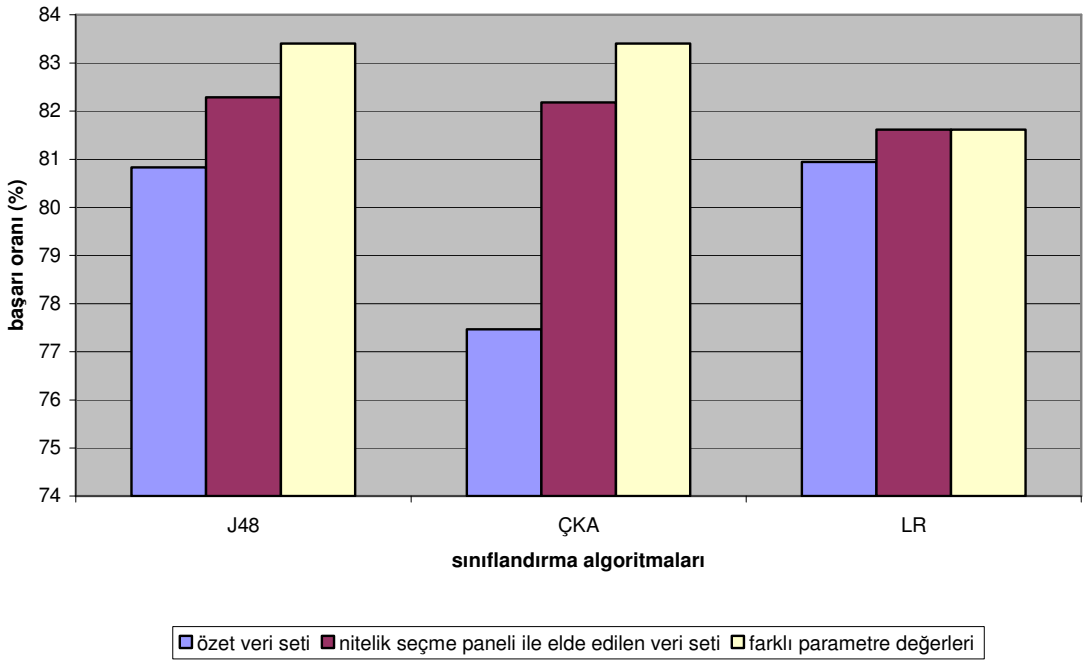
Momentum parametresinin 0,5 değerine karşılık gelen sonuçlar incelendiğinde (Çizelge 5.11), başarı oranında yaklaşık %0,2'lik bir artış sağlandığı görülmektedir (Çizelge 5.10).

ÇKA algoritmasının Öğrenme Katsayısı ve Momentum parametrelerinin farklı değerlerinin test edilmesi sonucunda, sınıflandırmadaki başarı oranı %83,4081 ve bu orana karşılık gelen Kappa istatistik değeri ise 0,5589 olarak bulunmuştur. Parametrelerde yapılan değişiklikler sonucunda J48 ve ÇKA algoritmalarının başarı oranlarında artış sağlanmıştır.

LR algoritmasında bulunan parametreler üzerinde değişiklik yapılamadığından, bu konu ile

ilgili herhangi bir yorum yapılamamaktadır.

Özetlenmiş veri setine uygulanan sınıflandırma algoritmalarının ürettiği sonuçlar, özetlenmiş veri setinin *Nitelik Seçme panelinde* indirgenmesi ile elde edilen değişkenlerin yeniden sınıflandırma algoritmalarına dahil edilmesi ile üretilen sonuçlar ve bu algoritmaların parametreleri üzerinde yapılan değişiklikler ile elde edilen sonuçlar Şekil 5.11'de gösterilmektedir.



Şekil 5.11 J48, ÇKA, LR algoritmalarının başarı oranları değişiminin karşılaştırılması.

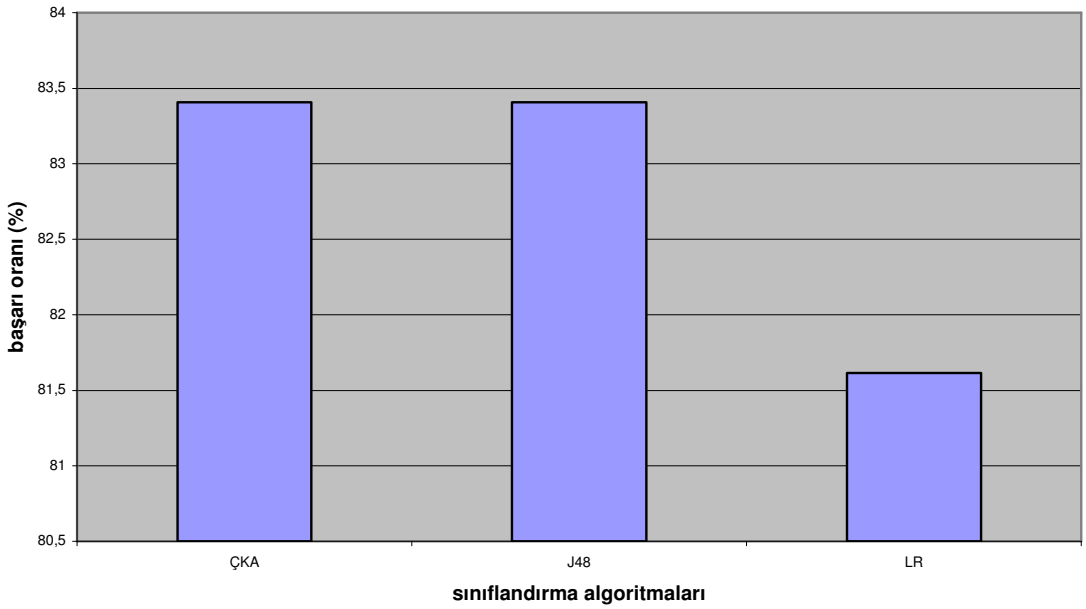
Şekil 5.11 incelendiğinde; özet veri seti için en iyi sınıflandırmanın LR algoritması ile gerçekleştirildiğini, *Nitelik Seçme paneli* verisi için J48 ile ÇKA'nın çok yakın sonuçlar ürettiğini, algoritmaların farklı parametre değerlerinin kullanılması sonucunda (LR algoritması hariç) ise J48 ve ÇKA algoritmalarının en yüksek başarı oranına ulaştıklarını görmekteyiz.

J48, ÇKA ve LR algoritmalarının Kredi Kartı Tipi veri setini sınıflandırmadaki başarı oranları ve bu oranlara karşılık gelen Kappa İstatistik değerleri Çizelge 5.12'de gösterilmektedir.

Çizelge 5.12 Kredi Kartı Tipi veri setine uygulanan sınıflandırma algoritmalarının başarı oranları ve Kappa istatistik değerlerinin karşılaştırılması.

Sınıflandırma Algoritmaları	**Başarı Oranları(%)**	**Kappa İstatistik Değerleri**
J48	83,4081	0,5617
ÇKA	83,4081	0,5589
LR	81,6143	0,4851

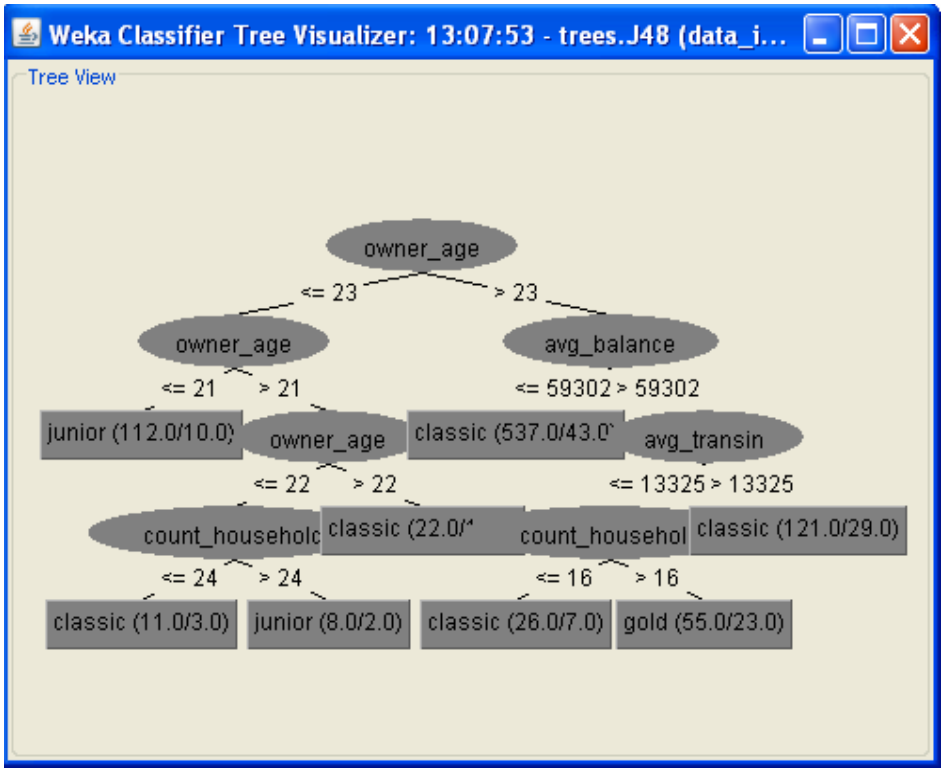
Sınıflandırma algoritmalarının başarı oranlarının grafiksel gösterimi Şekil 5.12'de yer almaktadır.



Şekil 5.12 J48, ÇKA ve LR Algoritmalarının başarı oranı değerlerinin grafiksel gösterimi.

Şekil 5.12 incelendiğinde; algoritmaların sınıflandırmadaki başarı oranlarının birbirlerine oldukça yakın sonuçlar verdikleri görülmektedir. Hatta J48 ağaç algoritması ile ÇKA algoritmasının aynı başarı oranlarına sahip oldukları görülmektedir. Ancak J48 algoritmasının Kappa İstatistik değeri, ÇKA algoritmasının Kappa İstatistik değerinden daha yüksek olduğu için modelin yorumlanmasında J48 algoritmasının kullanılmasına karar verilmiştir.

J48 algoritmasının oluşturduğu karar ağacı yapısının Weka'daki görünümü Şekil 5.13'de gösterilmektedir.



Şekil 5.13 Oluşturulan karar ağacının Weka'daki görünümü.

Şekil 5.13'de görülen J48 karar ağacı yapısı ile ilgili şu çıkarımlar yapılabilir:

- Yaşı 21 veya 21'den küçük olan müşteriler Junior kart sahibidir.
- Yaşı 23'den büyük, hesabında kalan parası 59302 Koruna*'dan fazla, hesabına yapılan işlemlerdeki para miktarı 13325 veya 13325 Koruna'dan az, ev giderleri için yaptığı harcamaların sayısı 16 veya 16'dan küçük olan müşteriler Classic kart; 16'dan büyük olan müşteriler ise Gold kart sahibidir.

Bu bilgiler ışığında banka, müşterileri için kredi kartı promosyonları düzenleyebilir.

5.2.2 Kredi Statüsü Veri Setinin Sınıflandırılması

Uygulamanın bu aşamasında, Kredi Statüsü veri setinin J48, ÇKA ve LR algoritmaları ile sınıflandırılması sonucu elde edilen sonuçları inceleyeceğiz. Kullanılan algoritmalarda kredilerin statüleri (A, B, C, D) bilgisi hedef değişken veya sınıf etiketi olarak belirlenmiştir.

* Çek Cumhuriyeti para birimidir.

5.2.2.1 Kredi Statüsü Veri Setinin J48, ÇKA ve LR Algoritmaları ile Sınıflandırılmasının Gerçekleştirilmesi

Uygulamanın bu bölümünde de J48, ÇKA ve LR algoritmaları tarafından kullanılacak olan eğitim ve test verilerinin ayrılması işlemi, sınıflandırma panelinde yer alan Çapraz Geçerleme Metodu'nun seçilmesi ile gerçekleştirilecektir.

J48, ÇKA ve LR algoritmalarının özet veriye uygulanması ile elde edilen çıkış değerleri Çizelge 5.13'de gösterilmektedir.

Çizelge 5.13 Özet veriye uygulanan sınıflandırma algoritmalarının ürettiği sonuçlar.

Sınıflandırma Algoritmaları	**Confusion Matrix**					**Başarı Oranı (%)**	**Kappa İstatistik Değerleri**
J48	a	b	c	d		87,0968	0,7623
	198	0	5	0	a=A		
	1	9	2	33	b=D		
	4	2	13	12	c=B		
	1	24	4	374	d=C		
ÇKA	a	b	c	d		84,6041	0,7183
	184	0	4	15	a=A		
	3	10	4	28	b=D		
	8	2	18	3	c=B		
	17	18	3	365	d=C		
LR	a	b	c	d		86,217	0,7532
	189	0	10	4	a=A		
	0	13	7	25	b=D		
	5	3	19	4	c=B		
	9	21	6	367	d=C		

5.2.2.2 Nitelik Seçme Panelinin Kredi Statüsü Veri Setine Uygulanması

Hangi bağımsız değişkenlerin bağımlı değişken üzerinde ne kadar açıklayıcı olduğunu belirlemek için özet veri seti N*itelik Seçme panelinde* işleme sokulmuştur. Elde edilen

sonuçların çıktısı Çizelge 5.14 ’de gösterilmektedir.

Çizelge 5.14 Veri setinde yer alan bağımsız değişkenlerin, bağımlı değişken üzerindeki açıklayıcılık değerleri.

Attribute Evaluator (supervised, Class (nominal): LoanStatus):

Information Gain Ranking Filter

Ranked attributes:		Ranked attributes:	
0,55207	CountLoanPay	0	ClientRegion
0,28362	LoanDuration	0	AvgStatement
0,26893	CountStatement	0	AvgOldagePension
0,21945	CountInterest	0	ClientAvgFreq
0,1894	CountHousehold	0	AvgTransIn
0,12911	LoanAmount	0	CountInsurance
0,04498	Balance	0	OrderAvgAmount
0,04283	AccDispStatus	0	AvgTransOut
0,00139	ClientSex	0	ClientAge
0	LengthOfCustomer	0	AvgInsurance
0	AvgHousehold	0	CountOldagePension
0	AvgInterestCredit	0	LoanPayments
0	ClientAccFreq		

Çizelge 5.14’de de açıkça görüldüğü gibi, bağımlı değişken üzerinde 9 bağımsız değişken; *CountLoanPay, LoanDuration, CountStatement, CountInterest, CountHousehold, LoanAmount, Balance, AccDispStatus* ve *ClientSex* bağımsız değişkenleri, Kredi Statüsü bağımlı değişkeni üzerinde açıklayıcılığa sahip değişkenlerdir. Sınıflandırmalar sonucu elde edilen değerler Çizelge 5.15’de gösterilmektedir. Bu 9 bağımsız değişkenin yeniden sınıflandırılması sonucu elde edilen değerler Çizelge 5.15’de gösterilmektedir.

Çizelge 5.15 Nitelik Seçme paneli ile indirgenen veriye uygulanan sınıflandırma algoritmaları sonuçları.

Sınıflandırma Algoritmaları	**Confusion Matrix**					**Başarı Oranı (%)**	**Kappa İstatistik Değerleri**
J48	a	b	c	d		89,4428	0,8025
	200	0	3	0	a=A		
	1	9	1	34	b=D		
	2	4	15	10	c=B		
	1	11	5	386	d=C		
ÇKA	a	b	c	d		89,2962	0,8033
	200	0	1	2	a=A		
	1	14	5	25	b=D		
	6	0	17	8	c=B		
	7	15	3	378	d=C		
LR	a	b	c	d		86,217	0,7532
	189	0	10	4	a=A		
	0	13	7	25	b=D		
	5	3	19	4	c=B		
	9	21	6	367	d=C		

Çizelge 5.15'te de görüldüğü üzere; J48 algoritmasının başarı oranı (Çizelge 5.13) yaklaşık %2,5 artışla %89'a, ÇKA'nın başarı oranı (Çizelge 5.13) ise yaklaşık %5'lik bir artışla yine %89 başarı oranına ulaşmıştır. LR algoritmasının başarı oranında ise herhangi bir değişiklik gözlenmemiştir.

5.2.2.3 Kredi Statüsü Veri Setine Uygulanan J48 ve ÇKA Sınıflandırma Algoritmalarının Parametrelerinin Değiştirilmesi

J48 algoritmasında yer alan Güven Faktörü ve MinNumObj parametrelerine atanan varsayılan değerler değiştirilerek sınıflandırmadaki başarı oranı artırılmaya çalışılmıştır. J48 algoritmasının farklı Güven Faktörü değerlerine karşılık gelen en yüksek 5 başarı oranı ve bu oranlara karşılık gelen Kappa istatistik sonuçları Çizelge 5.16'da gösterilmektedir.

Çizelge 5.16 Farklı Güven Faktörü değerlerine karşılık gelen Başarı Oranı ve Kappa İstatistik sonuçları.

Güven Faktörü	**Başarı Oranı(%)**	**Kappa İstatistik**
0,45	89,5894	0,8052
0,4	89,5894	0,8049
0,35	89,4428	0,8025
0,2	89,4428	0,8023
4	89,1496	0,7956

Çizelge 5.16'daki verilere göre; Güven Faktörü olarak 0,45 değeri, uygulamanın sonraki adımı için kullanılmaya uygun görülmüştür.

Uygulamanın bu adımında, Güven Faktörü 0,45 olarak seçilen J48 algoritmasının, farklı MinNumObj değerleri için ürettiği en yüksek 5 Başarı Oranı ve bu oranlara karşılık gelen Kappa İstatistik sonuçları incelenmiştir (Çizelge 5.17).

Çizelge 5.17 Farklı MinNumObj değerlerine karşılık gelen Başarı Oranı ve Kappa İstatistik sonuçları.

MinNumObj(Güven Faktörü=0,15)	**Başarı Oranı (%)**	**Kappa İstatistik**
2	89,5894	0,8052
4	89,1496	0,7956
3	89,0029	0,7954
5	88,563	0,7814
6	88,563	0,7811

MinNumObj parametresinin farklı değerleri için, J48 algoritmasının sınıflandırmadaki en yüksek başarı oranı ve bu orana gelen Kappa İstatistik değerinde bir artış elde edilememiştir.

J48 algoritmasının Güven Faktörü ve MinNumObj parametrelerinin farklı değerlerinin test edilmesi sonucunda, sınıflandırmadaki en yüksek başarı oranı %89,5894 ve bu orana karşılık gelen Kappa istatistik değeri ise 0,8052 olarak bulunmuştur.

ÇKA algoritmasında yer alan Öğrenme Katsayısı ve Momentum parametrelerine atanan varsayılan değerler değiştirilerek sınıflandırmadaki başarı oranı artırılmaya çalışılmıştır. ÇKA algoritmasının farklı Öğrenme Katsayısı değerlerine karşılık gelen en yüksek 5 başarı oranı

ve bu oranlara karşılık gelen Kappa İstatistik sonuçları Çizelge 5.18'de gösterilmektedir.

Çizelge 5.18 Farklı Öğrenme Katsayısı değerlerine karşılık gelen Başarı Oranı ve Kappa İstatistik sonuçları.

Öğrenme Katsayısı	**Başarı Oranı (%)**	**Kappa İstatistik**
0,15	90,4692	0,8237
0,16	90,4692	0,8226
0,17	90,3226	0,8192
0,21	90,176	0,8182
0,24	90,0293	0,8159

Öğrenme Katsayısının 0,15 değeri için elde edilen sonuçlar incelendiğinde (Çizelge 5.18), ÇKA algoritmasının sınıflandırmadaki başarı oranında %1'den fazla, başarı oranına karşılık gelen Kappa istatistik değerinde ise yaklaşık 0,02'lik bir artış elde gözlenmiştir(Çizelge 5.15).

ÇKA algoritması üzerinde etkin olabilecek bir başka parametre olan Momentum parametresinin farklı değerlerinin Öğrenme Katsayısı 0,15 olarak düzenlendikten sonra algoritmada kullanılması sonucu elde edilen değerler Çizelge 5.19'da gösterilmektedir.

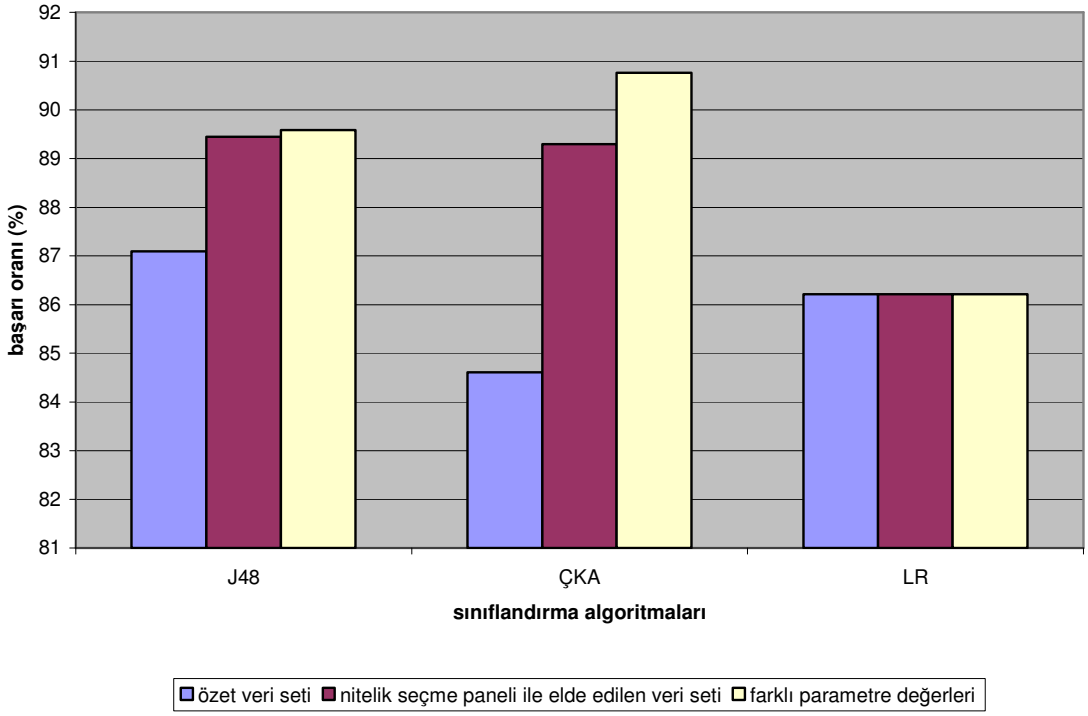
Çizelge 5.19 Farklı Momentum değerlerine karşılık gelen Başarı Oranı ve Kappa İstatistik sonuçları.

Momentum(Öğrenme Katsayısı = 0,15)	**Başarı Oranı (%)**	**Kappa İstatistik**
0,21	90,7625	0,8286
0,52	90,4692	0,825
0,2	90,4692	0,8237
0,25	90,4692	0,8226
0,24	90,4692	0,8224

ÇKA algoritmasının Öğrenme Katsayısı ve Momentum parametrelerinin farklı değerlerinin test edilmesi sonucunda, sınıflandırmadaki en yüksek başarı oranı %90,7625 ve bu orana karşılık gelen Kappa istatistik değeri ise 0,8286 olarak bulunmuştur.

LR algoritmasında bulunan parametreler üzerinde değişiklik yapılamadığından, bu konu ile ilgili herhangi bir yorum yapılamamaktadır.

Özet veri setine uygulanan sınıflandırma algoritmalarının ürettiği sonuçlar, özet veri setinin *Nitelik Seçme panelinde* indirgenmesi ile elde edilen değişkenlerin yeniden sınıflandırma algoritmalarına dahil edilmesi ile üretilen sonuçlar ve bu algoritmaların parametreleri üzerinde yapılan değişiklikleri ile elde edilen sonuçlar Şekil 5.14'de gösterilmektedir.



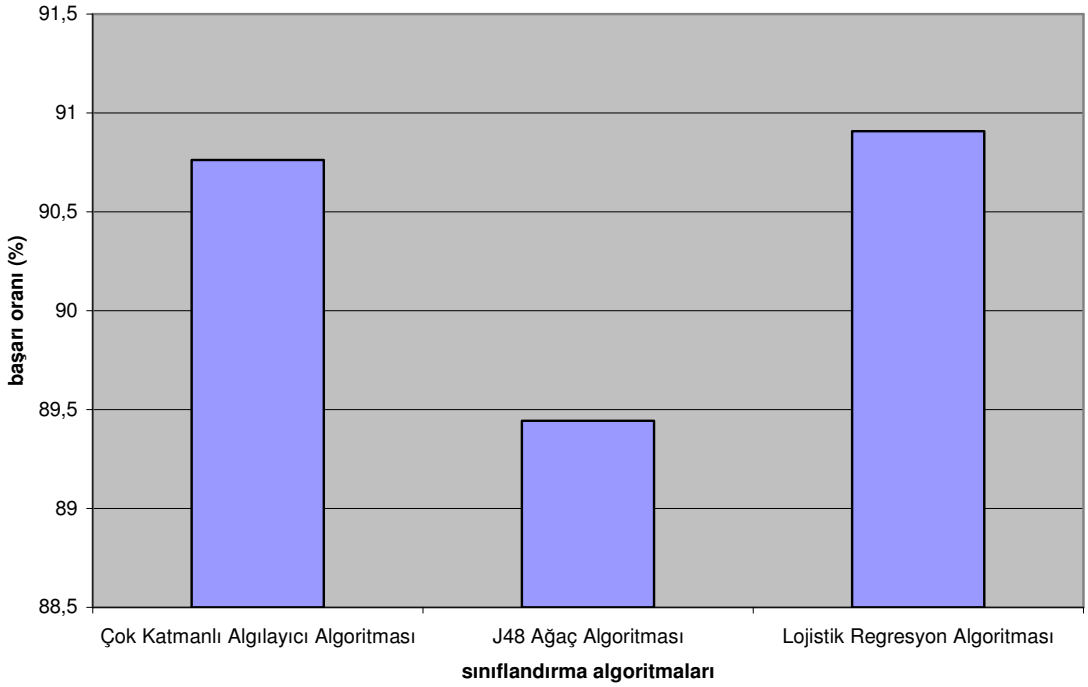
Şekil 5.14 Kredi Kartı Tipi veri setinin sınıflandırma algoritmalarındaki başarı oranlarının karşılaştırılması.

Şekil 5.14 incelendiğinde; özet veri seti için en iyi sınıflandırmanın J48 algoritması ile gerçekleştirildiği, Nitelik Seçme paneli verisi için J48 ile ÇKA'nın çok yakın sonuçlar ürettiği, algoritmaların farklı parametre değerlerinin kullanılması sonucunda (LR algoritması hariç) ise en yüksek başarı oranına ÇKA algoritmasının ulaştığını görmekteyiz.

J48, ÇKA ve LR algoritmalarının Kredi Statüsü veri setini sınıflandırmadaki başarı oranları ve bu oranlara karşılık gelen Kappa İstatistik değerleri Çizelge 5.20'de gösterilmektedir.

Çizelge 5.20 J48, ÇKA ve LR sınıflandırma algoritmalarının Başarı Oranları ve bu oranlara karşılık gelen Kappa İstatistik değerlerinin karşılaştırılması.

Sınıflandırma Algoritmaları	**Başarı Oranları (%)**	**Kappa İstatistik Değerleri**
J48	89,4428	0,8025
ÇKA	90,7625	0,8286
LR	90,9091	0,8321



Şekil 5.15 ÇKA ve LR algoritmalarının sınıflandırmadaki başarı oranları.

5.3 Müşteri Segmentasyonu

Uygulamanın bu aşamasında, farklı veri setleri için müşterileri segmentlere ayıracağız. Müşteri segmentasyonunda kullanılan algoritmaların başında kümeleme algoritmaları gelmektedir. Kümeleme algoritmaları içerisinde ise en çok tercih edilen algoritma K-Means algoritmasıdır. Uygulamanın bundan sonraki aşamasında, K-Means algoritmasının Weka paket programına uyarlanmış şekli olan Simple K-Means algoritması kullanılmıştır.

K-Means algoritması nümerik değerler ile çalışabilen bir algoritmadır. Ancak Weka'da yer alan Simple K Means algoritması hem nümerik hem de kategorik veriler ile çalışabilen bir algoritmadır.

Simple K-Means algoritmasında tanımlanan 2 parametre vardır.

1. numClusters parametresi: Algoritmada kullanılan küme sayısını gösterir.
2. seed parametresi: Her küme için rastgele atanan kayıt sayısını gösterir.

Müşteri segmentasyonu uygulaması Kredi Kartı Tipi ve Kredi Statüsü veri setleri için ayrı ayrı gerçekleştirildi.

5.3.1 Kredi Kartı Tipi Veri Seti için Kümeleme Analizi

Veri setine Simple K-Means algoritması uygulamandan önce veri üzerinde kutulama (binning) işlemi gerçekleştirilecektir. Kutulama işleminde veri, önceden belirlenen sayıda aralıklara bölünmektedir. Böylece kümeleme işlemi sonrasında elde edilen sonuçların yorumlanmasında kolaylık sağlanmaktadır. Müşteriler segmentleri ile ilgili genel sonuçlara ulaşabilmek için çalışmamızda verilerin 3 parçaya ayrılması uygun görüldü. Kutulama işlemi SQL sorguları ile gerçekleştirilebileceği gibi Weka paket programı ile de gerçekleştirilebilmektedir. Weka paket programında yer alan *Önişleme panelinde* yer alan *Discrete düğmesi* ile veri setinde yer alan kayıtlar yaklaşık 3 eşit parçaya ayrılmıştır. Çizelge 5.21'de veriye uygulanmış kutulama işlemi sonucu görülmektedir.

Çizelge 5.21 Kredi Kartı Tipi veri setine uygulanan kutulama işlemi.

Değişkenler	**Yapılan Kutulama İşlemi**
ClientAge	(18-31), (32-48), (49+)
OrderAvgAmount	(0-3344), (3345-6917), (6918+)
Balance	(22963-46194), (46195-56287), (56288+)
AvgTransIn	(1709-9667), (9668-13141), (13142+)
AvgTransOut	(624-4927), (4928-7571), (7572+)

Veri setinde yer alan kayıtlar üzerinde kutulama işlemi gerçekleştirildikten sonra elde edilen veriler kümeleme işlemine sokulmuştur. Simple K-Means algoritmasında yer alan ve küme sayısını ifade eden numClusters parametresinin 2, 3, 4, 5, 6, 7 değerleri için uygulama gerçekleştirilmiştir. Kümeleme sonuçları incelendiğinde kümeler arasındaki mesafenin en fazla olduğu küme sayısı değerinin, numClusters ifadesinin 3 değerinde sağlandığı görülmüştür. Çizelge 5.22'de 3 kümeye ayrılmış veri setinin Simple K-Means algoritması sonuçları görülmektedir.

Çizelge 5.22 Kredi Kartı Tipi veri setine uygulanan Simple K-Means algoritması sonuçları.

```
Scheme:       weka.clusterers.SimpleKMeans -N 3 -S 11
Instances:    892
Attributes:   6
              ClientAge
              OrderAvgAmount
              Balance
              AvgTransin
              AvgTransout
Ignored:
              CreditCardType
Test mode:    evaluate on training data

=== Model and evaluation on training set ===
kMeans
======
Number of iterations: 3
Within cluster sum of squared errors: 1857.0

Cluster centroids:

Cluster 0
Mean/Mode:  '(32-48)'  '(6918+)'  '(46195-56287)'  '(1709-9667)'  '(624-4927)'
Std Devs:         N/A          N/A               N/A              N/A          N/A
Cluster 1
Mean/Mode:  '(18-31)'  '(0-3344)'  '(22963-46194)'  '(1709-9667)'  '(4928-7571)'
Std Devs:         N/A          N/A               N/A              N/A          N/A
Cluster 2
Mean/Mode:  '(49+)'  '(3345-6917)'  '(56288+)'  '(13142+)'  '(7572+)'
Std Devs:         N/A          N/A           N/A          N/A          N/A
                                Clustered Instances
                                  0      346 ( 39%)
                                  1      259 ( 29%)
                                  2      287 ( 32%)
```

Simple K-Means algoritmasının veri setine uygulanması sonucunda 3 müşteri grubu elde edilmiştir. Müşteri gruplarının baskın özelliklerini gösteren bu kümeler Çizelge 5.23'de yer almaktadır.

Çizelge 5.23 Kredi kartı sahibi müşteri gruplarının baskın özellikleri

Nitelikler	**Grup 1 (Cluster 0)**	**Grup 2 (Cluster 1)**	**Grup 3 (Cluster 2)**
ClientAge	(32-48)	(18-31)	(49+)
OrderAvgAmount	(6918+)	(0-3344)	(3345-6917)
Balance	(46195-56287)	(22963-46195)	(56288+)
AvgTransIn	(1709-9667)	(1709-9667)	(13142+)
AvgTransOut	(624-4927)	(4928-7571)	(7572+)

Elde edilen çıktı incelendiğinde;

- 1.gruba ait müşterilerin hesaplarından yaptıkları otomatik ödeme işlemlerinin diğer gruptaki müşterilerden daha fazla olduğu görülmektedir.
- 2.grupta yer alan müşterilerin en genç müşteri grubunu oluşturduğu görülmektedir. Ayrıca bu müşteri grubu yapılan otomatik ödemelerin ve hesapta kalan para miktarının en az olduğu müşteri grubudur.
- 3.grupta yer alan müşterilerin ise en yaşlı müşteri grubunu oluşturdukları görülmektedir. Aynı zamanda bu gruptaki müşterileri hesaplarına/hesaplarından en çok para işlemi yapılan ve hesabında kalan parası en fazla olan müşteri grubunu oluşturmaktadır.

5.3.2 Kredi Statüsü Verisi için Kümeleme Analizi

Kredi Statüsü veri seti üzerinde de tıpkı Kredi Kartı Tipi veri setinde olduğu gibi kutulama işlemi gerçekleştirilmiştir. Yine veri setinde yer alan kayıtlar yaklaşık 3 eşit parçaya ayrılmıştır. Çizelge 5.24'de veriye uygulanan kutulama işlemi sonuçları görülmektedir.

Çizelge 5.24 Kredi Statüsü veri setine uygulanan kutulama işlemi.

Değişkenler	**Yapılan Kutulama İşlemi**
LoanAmount	(4980-79872), (79873-177774), (177775+)
LoanDuration	(12-30), (31-54), (55+)
CountStatement	(4-24), (25-42), (43+)
CountInterest	(10-30), (31-51), (52+)
CountHousehold	(0-10), (11-30), (31+)
CountLoanPay	(1-11), (12-23), (24+)

Veri setinde yer alan kayıtlar üzerinde kutulama işlemi gerçekleştirildikten sonra elde edilen veriler, kümeleme işlemine sokulmuştur. Simple K-Means algoritmasında yer alan ve küme sayısını ifade eden numClusters parametresinin 2, 3, 4, 5, 6, 7 değerleri için uygulama gerçekleştirilmiştir. Kümeleme sonuçları incelendiğinde kümeler arasındaki mesafenin en fazla olduğu küme sayısı değerinin, numClusters ifadesinin 3 değerinde sağlandığı görülmüştür. Çizelge 5.25'de 3 kümeye ayrılmış veri setinin Simple K-Means algoritması sonuçları görülmektedir.

Çizelge 5.25 Kredi Statüsü veri setine uygulanan Simple K-Means algoritması sonuçları.

```
Scheme:       weka.clusterers.SimpleKMeans -N 3 -S 10
Instances:    682
Attributes:   7
              LoanAmount
              LoanDuration
              CountStatetement
              CountInterest
              CountHousehold
              CountLoanPay
Ignored:
              LoanStatus
Test mode:    evaluate on training data

=== Model and evaluation on training set ===
kMeans
======
Number of iterations: 5
Within cluster sum of squared errors: 1520.0

Cluster centroids:
Cluster 0
Mean/Mode:  '(79873-177774)'  '(31-54)'  '(4-24)'  '(10-30)'  '(11-30)'  '(12-23)'
Std Devs:                N/A        N/A       N/A       N/A       N/A        N/A
Cluster 1
Mean/Mode:  '(4980-79872)'  '(12-30)'  '(43+)'  '(52+)'  '(31+)'  '(24+)'
Std Devs:                N/A        N/A      N/A      N/A      N/A      N/A
Cluster 2
Mean/Mode:  '(177775+)'  '(55+)'  '(25-42)'  '(31-51)'  '(0-10)'  '(24+)'
Std Devs:            N/A        N/A      N/A       N/A       N/A      N/A
                            Clustered Instances
                             0      290 ( 43%)
                             1      254 ( 37%)
                             2      138 ( 20%)
```

Simple K-Means algoritmasının veri setine uygulanması sonucunda 3 müşteri grubu elde edilmiştir. Müşteri gruplarının baskın özelliklerini gösteren bu kümeler Çizelge 5.26'da yer almaktadır.

Çizelge 5.26 Kredi Statülerine göre müşteri gruplarının baskın özellikleri

Nitelikler	**Grup 1 (Cluster 0)**	**Grup 2 (Cluster 1)**	**Grup 3 (Cluster 2)**
LoanAmount	(79873-177774)	(4980-79872)	(177775+)
LoanDuration	(31-54)	(12-30)	(55+)
CountStatement	(4-24)	(43+)	(25-42)
CountInterest	(10-30)	(52+)	(31-51)
CountHousehold	(11-30)	(31+)	(0-10)
CountLoanPay	(12-23)	(24+)	(24+)

Elde edilen çıktı incelendiğinde;

- 1.grupta yer alan müşteri grubunun; en az hesap özeti alan, ana parası üzerinden en az faiz alan ve en az kredi geri ödemesi yapan müşterilerden oluştuğu görülmektedir.
- 2.grupta yer alan müşteri grubunun; en çok hesap özeti alan, ana parası üzerinden en çok faiz alan, ev giderleri harcamasını ve kredi geri ödemesini en fazla yapan müşteri grubunu oluşturduğu görülmektedir.
- 3.grupta yer alan müşteri grubu ise miktar olarak en fazla, geri ödemesi ise en uzun krediyi alan müşterilerin oluşturduğu gruptur.

SONUÇ

Finans alanında önemli bir yere sahip olan bankacılık sektörü, rekabetin en yoğun yaşandığı sektörlerden biridir. Müşterilere sağlanan kredi kartları ile yine müşterilere verilen krediler, bankacılık sektörünün önemli yapı taşlarındandır. Bu iki alan aynı zamanda bankaların müşterilerini kaybetmemek ve yeni müşteriler kazanmak için uyguladığı stratejilerin temelinde yer almaktadır. Bu nedenle kredi kartı harcamaları ve sağlanan kredilerin analizinin gerçekleştirilmesi gerekmektedir. Kredi kartı harcamaları ve sağlanan kredilerin analizinin gerçekleştirilmesi, veri madenciliği teknikleri tarafından etkili bir şekilde yapılabilen analizlerdir.

Bu tezde; bazı sınıflandırma ve kümeleme algoritmalarının internet ortamında elde edilen bir Çek bankası verisi üzerinde uygulanması süreci incelenmiştir. Banka verisi, Kredi Kartı Tipi verisi ve Kredi Statüleri verisi olarak ikiye ayrılarak incelenmiştir.

Banka verisi, SQL Server 2000 programı ile veri önişleme aşamalarından geçirilerek veri madenciliği sürecinde kullanılmaya hazır hale getirilmiştir. Veri setinin modellenmesi süreci, Java programlama dili ile yazılmış olan Weka veri madenciliği programında yer alan sınıflandırma ve kümeleme tekniklerinin kullanılması ile gerçekleştirilmiştir.

Karar ağaçlarından J48 algoritması, yapay sinir ağlarından ÇKA ve regresyon analizinden LR algoritması veri seti üzerinde kullanılan sınıflandırma teknikleri ve algoritmalarıdır. Bu teknikler, Kredi Kartı Tipi verisi ile Kredinin Statüsü verisi üzerinde ayrı ayrı uygulanmıştır.

Kredi Kartı Tipi verisine uygulanan sınıflandırma algoritmaları, başarı oranları olarak birbirine son derece yakın sonuçlar vermişlerdir. Bu algoritmalardan J48 ve ÇKA algoritması, verinin sınıflandırılmasında aynı başarı oranını yakalamışlardır. Fakat modelin anlamlılığının bir ölçütü olan Kappa İstatistik değerinin J48 algoritmasında daha iyi sonuç vermesinden dolayı modellemede J48 algoritmasının kullanılması uygun görülmüştür.

Kredi Statüsü verisine uygulanan sınıflandırma algoritmaları içerisinde en iyi başarı oranı, az bir farkla da olsa LR algoritması tarafından gerçekleştirilmiştir.

Kümeleme algoritmaları içerisinde en çok tercih edilenlerden biri olan Simple K-Means algoritması, Kredi Kartı Tipi ve Kredi Statüsü verisine uygulanarak, her iki veri seti için de müşteri segmentleri oluşturulmuştur.

Kredi Kartı Tipi veri seti üzerinde uygulanan Simple K-Means algoritması sonucunda müşterilerin baskın özelliklerine göre 3 farklı müşteri grubu elde edilmiştir.

Bu müşteri grupları incelendiğinde; 1.gruba ait müşterilerin hesaplarından yaptıkları otomatik ödeme işlemlerinin diğer gruptaki müşterilerden daha fazla olduğu; 2.grupta yer alan müşterilerin en genç müşteri grubunu oluşturduğu; 3.grupta yer alan müşterilerin ise yaptıkları otomatik ödemelerin ve hesapta kalan para miktarlarının en az olduğu müşteri grubunu oluşturduğu görülmüştür.

Kredi Statüsü veri seti üzerinde uygulanan Simple K-Means algoritması sonucunda müşterilerin baskın özelliklerine göre de 3 farklı müşteri grubu elde edilmiştir.

Bu müşteri grupları incelendiğinde; 1.grupta yer alan müşteri grubunun; en az hesap özeti alan, ana parası üzerinden en az faiz alan ve en az kredi geri ödemesi yapan müşterilerden oluştuğu; 2.grupta yer alan müşteri grubunun; en çok hesap özeti alan, ana parası üzerinden en çok faiz alan, ev giderleri harcamasını en fazla yapan ve en fazla kredi geri ödemesi yapan müşteri grubunu oluşturduğu; 3.grupta yer alan müşteri grubunun ise miktar olarak en fazla ve geri ödemesi en uzun krediyi alan müşteri grubunu oluşturduğu görülmüştür.

Konu ile ilgili bundan sonra yapılacak çalışmalarda; veri madenciliği sürecinde kullanılan farklı sınıflandırma algoritmalarının başarı oranlarının karşılaştırılması ve elde dilen sonuçlara göre geçerli bir modelin oluşturulması gerçekleştirilebilir.

KAYNAKLAR

Berry, Michael J.A. ve Linoff, Gordon., (2000), Mastering Data Mining, The Art and Science of Customer Relatioship Management, New York, Wiley Publishing.

Berson, Alex., Smith, Stephan. ve Thearling, Kurt., (2000), Building Data Mining Applications for CRM, McGraw Hill.

Bounsaythip, Catherine., Runsala, Esa Rinta., (2001), Overview of Data Mining for Customer Behavior Modeling, Vtt Information Technology Research Report.

Brown, Stanley A., (2000), Customer Relationship Management: A Strategic Imperative in the World of E-Business, John Wiley&Sons.

Child, Peter., Dennis, Robert J., Gokey, Timothy C., McGuire, Tim I., Sherman, Mike., ve Singer, Marc., "Can Marketing Regain Touch?", www.crm-forum.com

Freeman, M., "The 2 Customer Lifecycles", (1999), Intelligent Enterprise, Vol.2/16

Groth, Robert., (2000), Data Mining, Building Competetive Advantage, New Jersey, Prentice Hall.

Hair, Joseph F., Anderson, Rolph., Tathma, Ronald. ve Black, William. (1998), Multivariate Data Analysis, 5.Baskı, New Jersey.

Han, Jiawei., ve Kamber, Micheline., (2001), Data Mining:Concepts and Techniques, Morgan Kaufmann Publisher, New York.

Kovalerchuk, Boris ve Vityaev, Evgenii, (2000), Data Mining in Finance, US, Kluwer Academic Publishers.

Lanquillon, Carsten., (1999), "Dynamic aspecrs in neural classification", International Journal of Intelligent Systems in Accounting, Finance and Management, Vol.8.

Manly, Bryan., (1989), Multivariate Statistical Methods, Chapman&Hill, London.

Musaoğlu, C., (2003), Customer acquisition and retention modeling in consumer finance sector using data mining. Thesis Study, Boğaziçi Press, İstanbul.

Orhunbilge, Neyran., (1996), Uygulamalı Regresyon ve Korelasyon Analizi, İ.Ü.İşletme Fakültesi Yayınları.

Özdamar, Kazım., (1999), Paket Programlar ile İstatistiksel Veri Analizi (Çok Değişkenli Analizler), 2.Baskı, Kaan Kitapevi, Eskişehir.

Özmen, Şule, (2003), Ağ Ekonomisinde Yeni Ticaret Yolu: E-Ticaret, İstanbul, İstanbul Bilgi Üniversitesi Yayınları.

Child, Peter., Dennis, Robert J., Gokey, Timothy C., McGuire, Tim I., Sherman., ve Singer, Marc, (2003), "Can Marketing Regain Touch?", www.crm-forum.com.

Ramamoorti, Sridhar., Bailey, Andrew D. ve Traver, Richard O., (1999), "Risk assessment in internal auditing: a neural network approach", International Journal of Intelligent Systems in Accounting, Finance and Management, Vol.8.

Richard, A.Johnson ve Dean, W.Wichern., (1988), Applied Multivariate Statistical Analysis, 2nd Edition, Prentice Hall, New Jersey.

Rygielski, Chris., Wang, Jyun-Cheng., ve Yen, David C., (2002), "Data mining techniques for customer relationship management", Technology in Society, Vol.24.

Shaw, Michael J., Subramaiam, Chandrasekar., Tan, Gek Woo., ve Welge, Michael E., (2001), "Knowledge management and data mining for marketing", Decision Support Systems, Vol.31.

SPSS Inc**.** Answer Tree 2.0 User's Guide

Stefansowski, Jery., ve Wilk, Szymon., (1997), "Evaluating business credit risk by means of approach-integrating decision rules and case-based learning**",** International Journal of Intelligent Systems in Accounting, Finance and Management, Vol.10.

Vojinovic, Zoran., Kecman, Vojislav., ve Seidel, Rainer., "A data mining approach to financial series modelling and forecasting", (2001), International Journal of Intelligent Systems in Accounting, Finance and Management, Vol.10.

Yazıcı, Ayşe Canan., Öğüş, Ersin., Ankaralı, Seyit., Canan, Sinan., Ankaralı, Handan., ve Akkuş, Zeki., (2007), "Yapay Sinir Ağlarına Genel Bakış: Derleme", Türkiye Klinikleri J Med Sci, 27:65-71.

Yeo, Ai Cheo., Smith, Kate A., Willis, Robert J. ve Brooks, Malcolm., (2001), "Clustering techniques for risk classification and prediction of claim costs in automobile industry", International Journal of Intelligent Systems in Accounting, Finance and Management, Vol.10.

ÖZGEÇMİŞ

Doğum tarihi	23.02.1983	
Doğum yeri	AĞRI	
Lise	1995-1998	Kocasinan Lisesi
Lisans	1999-2004	İstanbul Üniversitesi Fen Fakültesi Fizik Bölümü