T.C.
MUĞLA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

İSTATİSTİK VE BİLGİSAYAR BİLİMLERİ ANABİLİM DALI

VERİTABANLARI ÜZERİNDE VERİ MADENCİLİĞİ UYGULAMASI

YÜKSEK LİSANS TEZİ

HÜSEYİN GÜRÜLER

MUĞLA 2005

Dizin Terimleri:(Dizin terimleri listelerinden seçiniz. İmleci dizin terimini girmek istediğiniz kutucuğa getiriniz.Kutucuğun yanındaki linke tıklayınız. Gelen alfabetik listeden uygun harfi seçiniz. Aradığınız terimi listede tarayıp bulduğunuzda tıklayınız. Terim uygun kutucuğa yerleşecekti Uyarı: Dizin terimi seçmek için yapılan ilk tıklamada Tez Tarama sayfası açılabilmektedir. Kapatıp liste linkini ikinci kez tıklatın sorun çözülecektir.)

Türkçe Dizin Terimleri		İngilizce Dizin Terimleri	
Veri madenciliği	Türkçe	Data mining	İngilizce
Sınıflandırma	Türkçe	Classification	İngilizce
Karar ağacı	Türkçe	Decision tree	İngilizce
	Türkçe		İngilizce
	Türkçe		İngilizce

Önerilen Dizin Terimleri:(YÖK Dizin terimleri listelerinde bulamayıp önerdiğiniz terimler)

Türkçe	İngilizce
Veritabanlarında bilgi keşfi	Knowledge discovery on databases

Tezin Metin Formatı Dışındaki Ekleri : (Aynı türden 1'den çok dosyanız varsa ilgili kutuda dosya adlarını noktalı virgül (;) ile ayırınız

Resim: ☐ Dosya adı:

Harita: ☐ Dosya adı:

Görüntü: ☐ Dosya adı:

Ses: ☐ Dosya adı:

Program: ☐ Dosya adı:

Diğer: ☐ Lütfen Belirtiniz: Dosya adı:

Kısıtlama Yok : ☐ Kısıtlama Var: ☐ Kısıtlama Bitiş Tarihi: (gg/aa/yyyy)

Proje desteği aldıysa Proje no :

Tarih: .....07/10/2005.....

İmza .....

Bu belgenin İnternet adresi : http://www.yok.gov.tr/tez/veri_giris5.htm

Dr. Mehmet KARAHASAN danışmanlığında Hüseyin GÜRÜLER tarafından hazırlanan bu çalışma, 26/09/2005 tarihinde aşağıdaki jüri tarafından İstatistik ve Bilgisayar Anabilim Dalı'nda yüksek lisans olarak oybirliği ile kabul edilmiştir.

Başkan	: Prof.Dr. Mübariz EMİNOV	İmza	:
Üye	: Yrd.Doç.Dr. Mahmut TENRUH	İmza	:
Üye	: Dr. Mehmet KARAHASAN	İmza	:
Üye	:	İmza	:
Üye	:	İmza	:

ÖNSÖZ

Bu tez çalışması, üniversitenin bünyesindeki elektronik ortamda saklı öğrenci verileri üzerinde veri madenciliği içerisinde bir sınıflandırma tekniği olan karar ağaçları kullanılarak Muğla Üniversitesi öğrencilerinin lisans eğitimlerinde onları başarıya götüren kişisel özellikleri bir profil olarak keşfetmek amacıyla gerçekleştirilmiştir.

Çalışmanın yönetilmesinde, değerli bilgi ve yardımlarıyla büyük ölçüde katkı sağlayan ve ilgisini esirgemeyen sayın hocam Dr. Mehmet KARAHASAN'a teşekkür etmeyi zevkli bir görev bilirim.

Değerli fikirlerinden her zaman istifade ettiğim Sayın hocalarım; Prof.Dr. Mustafa DİLEK, Prof.Dr. Mübariz EMİNOV, Yrd.Doç.Dr. O. Nuri YİĞİTBAŞI ve Dr. Ayhan İSTANBULLU'ya teşekkürü bir görev bilirim.

Muğla Üniversitesi öğrenci verilerini edinmem ve düzenlememde yardımcı olan Sayın Okt. Kürşat KURT ve onun nezdinde Bilgi İşlem Dairesi Başkanlığı ve Öğrenci İşleri Dairesi Başkanlığı'na çalışmalarımda yapmış oldukları değerli katkılarından ötürü teşekkürü bir borç bilirim.

Bu çalışmanın, üniversitenin en değerli kaynağı olarak nitelendirilebilecek öğrencileri tanımlama ve tanımamızda fayda sağlayacağına ve bu anlamda daha hızlı ve daha gelişmiş araçların araştırılmasında katkı sağlayacağı inancındayım.

Hüseyin GÜRÜLER
MUĞLA
2005

İÇİNDEKİLER DİZİNİ

VERİTABANLARI ÜZERİNDE VERİ MADENCİLİĞİ UYGULAMASI

(Yüksek Lisans Tezi)

Hüseyin GÜRÜLER

MUĞLA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

2005

ÖZET

Bu tezin amacı, Muğla Üniversitesi öğrencilerinin eğitimdeki başarı profilini ortaya çıkaran, bilgi tabanlı bir sistemin geliştirilmesi ve ham veri şeklindeki mevcut öğrenci kayıtlarının, üniversite öğrencilerini tanımada fayda sağlayacak bilgilere dönüştürmektir.

Veri madenciliği, büyük miktarda veri içerisinden anlamlı ve yararlı bağıntı ve kuralların bilgisayar programları kullanarak aranması ve analizidir. Veritabanlarında bilgi keşfi ise ham veriden bilgi elde etmeye kadar veri madenciliğini de içine alan geniş bir süreçtir.

Çalışmada kullanılan Muğla Üniversitesi öğrenci verileri, Öğrenci İşleri Daire Başkanlığı'nın izni ile alınmıştır. Bilgi keşfi işlemleri, İktisadi ve İdari Bilimler Fakültesi 1995 yılı ve sonrası öğrenci verileri üzerinde gerçekleştirilmiştir. Çalışmada, bilgi keşfi sürecini veritabanı yönetim sistemi ile bütünleştiren bir araç olarak MÜKÜP (Muğla Üniversitesi Öğrenci Bilgi Keşif Ünitesi Programı) geliştirilmiş ve bu öğrenci verileri üzerinde uygulanmıştır.

Çalışma iki kısımdan oluşmaktadır. İlk olarak gerçekleştirilen veritabanı sorgulamaları ile üniversite öğrencilerine ait demografik bilgileri içeren bir bilgi dağılımı elde edilmiştir. Aynı zamanda keşif sürecinin veri hazırlama ve model oluşturma bölümlerinde uygun kararlar alabilmek için ihtiyaç duyulan eldeki veriyi tanıma gerçekleştirilmeye çalışılmıştır. İkinci bölümde bilgi keşfi süreci yer almaktadır. Burada öğrencilerin kişisel verileri ile üniversitede elde ettikleri başarı verileri

birleştirilerek bir sınıflandırma modeli gerçekleştirilmiştir. Sınıflandırma modeli, üniversite öğrencilerinin eğitimdeki başarılarına hangi kişisel özellikleri en çok etki ettiğini araştırmaktadır. Model oluşturma ve gösteriminde bir veri madenciliği algoritması olan karar ağaçları kullanılmıştır.

Araştırma sonucunda, öğrencilerin üniversiteye kayıt olma türü ve aile gelir düzeylerinin eğitimdeki başarılarında etkili olduğu görülmüştür.

Anahtar Kelimeler: Veritabanlarında Bilgi Keşfi, Veri Madenciliği, Sınıflandırma, Karar Ağaçları

Sayfa adedi :124
Tez yöneticisi :Dr. Mehmet KARAHASAN

DATA MINING APPLICATION ON DATABASES

(M. Sc.Thesis)

Hüseyin GÜRÜLER

MUĞLA UNIVERSITY
INSTITUTE of SCIENCE and TECHNOLOGY

2005

ABSTRACT

The goal of this thesis is to develop a knowledge based system that produces the success profile of the students of Mugla University and to transform student's records that are in form of raw data into information that helps us better understand the students.

Data mining is extracting and analyzing meaningful and beneficial rules and relations from a large set of data using computer programs. Knowledge discovery on databases is a process that spans from data mining to extracting information from raw data.

In this study, Mugla University student data were used with permission of the Office of Student Affairs. Student data used for knowledge discovery process was chosen from Faculty of Economical and Administrative Sciences that are on and after 1995. In this study, MUKUP (Mugla University Student Knowledge Discovery Unit Program) was developed and applied on the student data.

This study has two parts. At first, using database queries, a knowledge distribution has been achieved that consist of student demographic data. At the same time, the student data was investigated which is needed to get appropriate decisions during the data preparation and modeling stages of discovery. Second part consists of knowledge discovery process. Individual records of students and success data of students are

combined and a classification model is developed classification model investigates which personal characteristics of the students has the most effect on student access. For modeling and representation, a data mining algorithm called decision trees is used.

Research results show that registration type and family income of the students affect their educational success.

Key Words: Knowledge Discovery on Databases, Data Mining, Classification, Decision Trees

Page number : 124
Adviser : Dr. Mehmet KARAHASAN

ŞEKİLLER DİZİNİ

TABLOLAR / ÇİZELGELER DİZİNİ

Tablo No Sayfa No

SEMBOLLER ve KISALTMALAR

ADO	ActiveX Data Objects
CART	Classification and Regression Trees
CHAID	Chi-squared Automatic Interaction Detection
DSO	Decision Support Objects
DTS	Data Transformation Services
İİBF	İktisadi ve İdari Bilimler Fakültesi
K-NN	K- Nearest Neighbor
MBR	Memory Based Reasoning
MDAC	Microsoft Data Access
MDT	Microsoft Decision Trees
MÖ	Makine Öğrenimi
MÜKÜP	Muğla Üniversitesi Öğrenci Bilgi Keşif Ünitesi Programı
NASA	National Aeronautics and Space Administration
OLAP	Online Analytical Processing
OLE DB	Object Linking and Embedding Database
OLTP	Online Transaction Processing
ÖSS	Öğrenci Seçme Sınavı
ÖSYM	Öğrenci Seçme ve Yerleştirme Merkezi
SDK	Software Development Kit
SQL	Structured Query language
T.C.	Türkiye Cumhuriyeti
VA	Veri Ambarlama
VBK	Veritabanlarında Bilgi Keşfi
VBScript	Visual Basic Scripting Edition
VM	Veri Madenciliği
YZ	Yapay Zeka

1.GİRİŞ

Günümüz modern insanının her alışverişinde, her bankacılık işleminde, her telefon konuşmasında kaydedilen; uzaktaki algılayıcı ve uydulardan toplanan; resmi ve özel işletmelerin yönetiminde yapılan işlemler sonucunda saklanan veriler her an artmaktadır. Bir işletme veya kurum için verileri toplama ve saklama amaçlarından birisi, bu verileri kullanarak bulunduğu sektördeki etkinliğini artıracak yönde bilgi elde etmektir. Bu nedenle zamanın geçmesi ile beraber daha fazla veri toplamak durumunda kalınır. Geçen son yirmi yılı aşkın bir süredir veri toplama araçlarındaki ilerlemeler sayesinde artan sayıda iş verisi, elektroniksel olarak depolanabilmekte ve bunun daha hızlı oranlarda artacağı beklenmektedir. Şu an donanım ve veritabanı teknolojileri etkili ve ucuz bir şekilde uygun veri depolama ve erişimine izin vermektedir. Diğer taraftan veriyi anlama yeteneği, veriyi bir araya getirme ve toplama becerisinin oldukça gerisinde kalmaktadır. Doğru karar vermek ve yönetmek büyüyen hacimdeki veriyi etkili analiz edebilmeye bağlıdır. Şu anda dünya çapındaki işletmeler, büyük çaptaki veriyi sadece geleneksel istatistik yöntemleri ile analiz etmenin zaman alıcı olduğu ve yönetiminin zor olduğunu düşünmektedirler (Akpınar, 2000). İşte bu nedenlerle "Büyük miktardaki veriden anlamlı ve kullanışlı desen ve kuralları keşfetmek için araştırma ve analiz etme süreci" (Berry vd., 2000) olarak tanımlanan VBK (Veritabanlarında Bilgi Keşfi) hayata girmiş durumdadır.

Pazarda lider olabilmek; müşteri memnuniyetini derinlemesine anlamak, hızlı bir şekilde müşteriye yönelik yeni stratejileri benimsemek ve gerçek ölçütlerle müşteri performansını değerlendirme kapasitesine bağlıdır. Verilerin miktar ve tür bakımından çeşitliliğinin artması, analizlerin daha hızlı yapılması gereği ve sonuçta anlamlı ve eyleme yönelik bilgiler ortaya çıkarılması, değişen piyasalardaki yoğun rekabet ortamının bir gereğidir. Şu andaki teknolojik gelişmeler ile milyarlarca bitlik verilerin depolanabildiği veri ambarları ve bunlardan çok hızlı bir şekilde anlamlı bilgiler elde edebilmek için kullanılabilecek yazılımlar sayesinde bu amaç gerçekleştirilmeye çalışılmaktadır. Burada veri madenciliği ürünlerinin ham malzemeyi sınıflandırma, bilginin cevherini elde etmedeki kabiliyetinden faydalanılmaktadır (Özmen, 2002).

Araştırmacıların, geniş hacimli ve çok dağınık veri kümeleri üzerinde yapmış oldukları çalışmalar sonucunda VM (Veri Madenciliği) ve bilgi keşfi (data mining & knowledge discovery), özellikle elektronik ticaret, tıp ve eğitim alanlarındaki uygulamalarda yeni ve temel bir araştırma sahası olarak ortaya çıkmıştır (Vahaplar vd, 2002).

Üniversiteler, bilginin üretimi ve yayılmasında üstlendiği çok önemli rolünün yanında öğrenciler için eğitim hizmeti sağlayan kurumlardır. Üniversiteler, kendi öğrenci verileri ile karar verme süreçlerini birleştirebilmek için yeni çözümler keşfetmektedirler. Bunlardan birisi de geçmiş öğrenci verilerinden faydalanılarak edinilmiş bilgiler ile desteklenen rehberlik sistemidir. İyi bir öğrenci ilişkisel yönetim ve yüksek öğrenci başarısını elde edebilmek için üniversitede öğrencilerinin öğrenim süreleri boyunca gerçekleştirilen akademik rehberlik, öğrencinin başarısına olumlu yönde katkıda bulunabilir (Güvenç, 2001).

Bu çalışmanın amacı, Muğla Üniversitesi'nde öğrenci performansını değerlendirme ve geliştirmede veri madenciliği yöntem ve araçlarını kullanılarak veri yönelimli bir yaklaşım önerisi sunmaktır. Bu araştırmada pilot olarak İİBF (İktisadi ve İdari Bilimler Fakültesi) öğrenci verileri üzerinde çalışmalar yapılmıştır.

Öğrenci performansını değerlendirmede birinci adım olarak, öğrencilerin geçmiş kayıtları üzerinde çeşitli veritabanı sorguları gerçekleştirilmiştir. Bu sorgular neticesinde, çalışmaya katkı sağlayacak, öğrencilere ait çeşitli demografik bilgiler, üniversite öncesi eğitim bilgileri ve üniversitedeki eğitimleri süresince oluşturdukları bazı bilgiler elde edilmiştir. İkinci adımda, keşfe yönelik bir çalışma olarak öğrenci sınıflandırma gerçekleştirilmiştir. Öğrenci sınıflandırmasında, ikinci bölümde anlatılan karar ağacı tekniği kullanılmıştır. Burada kategorik bir değişken, hedef değişken olarak belirtilir ve yeni kayıtları önceden belirlenmiş sınıflara yerleştirmek için tahminleyici model geliştirilir. Öğrenciler, sınıflandırmada kişisel geçmişleri ve dönem ortalamalarına dayalı olarak başarılı veya başarısız olarak sınıflandırılır. Muğla Üniversitesi'nde geçmişe dair veriler üzerinde çalışmak; şu andaki durumu doğru bir şekilde anlamak ve yeni gelen öğrencilerin gelecekteki başarı durumlarını tahmin edebilmeyi sağlamak içindir.

Çalışmada kullanılan öğrencilere ait veriler, üniversitenin Bilgi İşlem Dairesi ve Öğrenci İşleri Dairesi izni ile alınmıştır. Alınan bu ham veriler, yerel bir veritabanında tutulduktan sonra veriler üzerindeki tüm bilgi keşif süreci işlemleri, bu iş için geliştirilen MÜKÜP (Muğla Üniversitesi Öğrenci Bilgi Keşif Ünitesi Programı) ile gerçekleştirilmiştir.

Tezin geri kalan kısımlarında; öncelikle VBK ve VM'nin altında yatan temel kavramlar tanıtılarak, süreç hakkında temel anlayış kazandırmak hedeflenmiştir. Burada VBK süreci ve bunların altında bulunan görevlerin ayrıntılı açıklaması, VM'nin uygulama alanları ve değişik sektörlere olan etkisi ve etkinlikleri tartışması, uygun yazılım araçları ve teorik teknikler ve gelecek yönleri sunulmuştur. Üçüncü bölüm, çalışmanın çekirdeğini oluşturmaktadır. Bu bölüm, VM tekniklerinin ve diğer VBK sürecine ait görevlerin çalışma içerisinde nasıl uygulandığı konularını yani uygulama metodolojisini içermektedir. 4. bölüm tüm analizlerde elde edilen bulguları vermektedir. Son bölümde, sonuçlar ve gelecek çalışmalar için tavsiyeler sunulmuştur.

2. KAYNAK ÖZETLERİ

2.1. Bilgi Keşfi Çatısı

Son 10 yılda, veri toplama ve toplanan verileri saklama teknolojileri büyük bir gelişme kaydetmiştir. Ticari faaliyetlerin otomasyonu ve barkod teknolojisindeki gelişmeler, verinin oluştuğu anda veritabanlarında saklanmasına imkan vermiştir. Gerçekte, her 20 ayda bir dünyadaki veri miktarının iki katına çıktığı tahmin edilmektedir (Frawley vd., 1991). Veri artışındaki büyümenin örnekleri, tüm sektörlerde görülmektedir. Kredi kartı kullanımı, tıbbi test sonuçları, telefon konuşmaları, süper marketlerde bir kerede satın alınan ürünler gibi en basit hareketler bile bilgisayar ortamına kaydedilmektedir.

Bilimsel ve resmi kuruluşlarda bulunan veritabanları da hızla büyümektedir. NASA, şimdiden analiz edebileceği veriden daha fazlasını veritabanlarında saklamaktadır (Fayyad vd., 1996). Örnek olarak, dünya gözlem uyduları bir günde bir terabayt veri üretmektedir. Bir günde üretilen resimleri, her resme bir saniye ayırarak bir kişi baksa idi; yalnızca resimlere bakma işlemi, o kişinin hafta sonları ve geceleri de çalışmak suretiyle bir kaç yılını alacaktır.

Farklı sektörlerde bulunan büyük veritabanları, içinde değerli bilgileri barındıran ve bu bilgilerin etkili bilgi keşif teknikleriyle ortaya çıkarılacağı bir veri madeni olarak görülebilir; ancak bu büyüklükteki veriyi analiz ederek anlamlı örüntüler elde etmek, insan yeteneğinin ve günümüz ilişkisel veritabanı teknolojilerinin sınırlarını aşmaktadır (Vahaplar ve İnceoğlu, 2002).

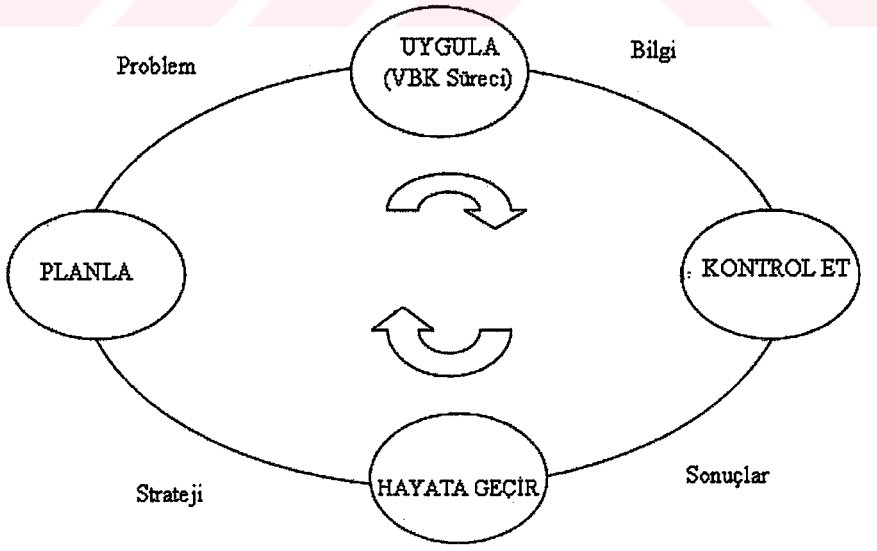
Veritabanı sistemlerinin artan kullanımı ve hacimlerindeki bu olağanüstü artış karşısında toplanan bu çok büyük miktarda ham veriyi özümseme ve yorumlama alanındaki yetersizlik, organizasyonları bu verilerden nasıl faydalanılabileceği düşüncesi ile karşı karşıya bırakmıştır. Bu bağlamda 1990'lı yıllara kadar daha çok verinin toplanması ve depolanması ile ilgilenilmiş. 1990'dan itibaren ise veri ambarlarının kullanımının yaygınlaşması ve veri analizi ön plana çıkmıştır (Bulun vd, 2003).

Eski nesil istatistiksel analiz teknikleri ve dosya yönetim araçları, artık büyük miktarlarda verinin analizinde uygun değildir. Bu nedenle, yerlerini VBK ve VM olarak

adlandırılan yeni nesil teknik ve araçlara bırakmaktadırlar. Bu yeni teknik ve araçlar veri analizinde değerli bilgileri keşfetmek için insana akıllı bir şekilde yardım ederler (Yurtsever, 2002).

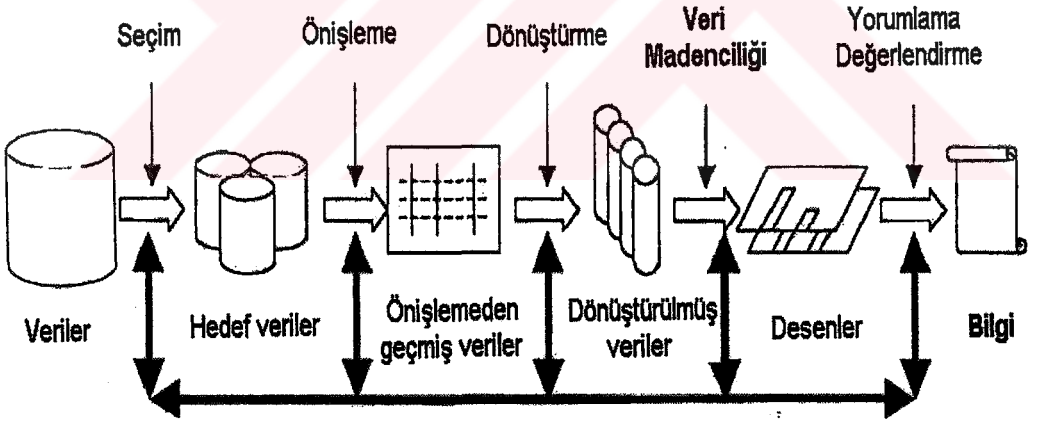
Aktif bir araştırma alanı olarak VBK teknolojisi, çok büyük hacimdeki verileri tam ya da yarı otomatik bir biçimde analiz eden yeni kuşak araç ve tekniklerin üretilmesi ile ilgilenen son yılların gözde araştırma konularından biridir. VBK süreci bir bütün olarak veri üzerindeki geçerli, yeni, kullanışlı ve son olarak anlaşılabilir desenleri keşfetmeyi tanımlamaktır (Agrawal vd., 1993; Brachman vd., 1996; Fayyad vd., 1996). Burada ***bilgi***, potansiyel olarak ilginç ve faydalı olan veriler arasındaki ilişki; ***keşif*** ise, gizlenmiş veya daha önceden bilinmeyen nesnelerin bulunması anlamına gelmektedir.

VBK ile VM arasında devam eden bazı yanlış anlamalar vardır. Sıklıkla birbirleri yerine kullanılmaktadır. VBK terimi, düşük seviyeli veriyi yüksek seviyeli bilgiye çevirme sürecinin (verinin seçimi, temizlenmesi, indirgenmesi, VM ve çıkan sonuçların değerlendirilmesi) tümünü ifade etmek için kullanılır. VM ise genellikle gözlenen veriden desenleri veya modelleri çıkarmada kullanılır. VM, VBK sürecinin çekirdeğini oluşturmasına rağmen, süreç boyunca sarf edilen tüm çabalar içinde ufak bir parçasını (%15-25 arasında) oluşturan önemli bir adımdır (Fayyad vd., 1996; Berson vd., 1997).



Şekil 2.1 Bilgi keşfi (Planla-Uygula-Kontrol Et-Hayata Geçir) çevrimi

Şekil 2.1'de bilgi keşfi çevrimi görülmektedir. Çevrim, *planla* aşaması ile başlar. Burada tam olarak üzerinde çalışılacak işle ilgili hedefler ve bunların uygulama alanları buna karşılık elde bulunan imkanlar ve olası problemler öğrenilir ve tanımlanmaya çalışılır. Projenin kapsamına burada karar verildiğinden bu aşama çok önemlidir. Sonra *uygula* aşaması gelir. Şekil 2.1'de gösterilen VBK sürecini içerir. Geniş veritabanından hedef veri kümesi seçilir. Bundan sonra temizleme, önişleme ve temizleme ile ilerideki dönüşüm; üzerinde çalışılacak doğru veri kümesini oluşturur. VM fonksiyonları ve algoritmaları seçilerek bu veri kümesinde işletilir. Bu sürecin sonunda ortaya çıkan, ticari kar olarak çevrilebilir ve değerlendirilebilir, keşfedilmiş bilgidir. Bu veri içerisinde bulunan ilişkileri ve desenleri içeren bilgi, *kontrol* aşamasına girdi olarak kullanılır. Buradaki analizler ile bilginin proje kapsamında uygulanabilirliği test edilir. Sonuçlar üretilir. Sonuçlar ilgi çekici ve tatmin edici bulunursa önerilen çözüm uygulanmak suretiyle son aşama olan *hayata geçirilir*. Sonuçta bu çevrim süreklidir. Yeni ve ilgili problemler zaman içerisinde gözükürse, VM araçlarının yardımı ile mekanizma, çözümleri bulmaya çalışacaktır (Güvenç, 2001).



Şekil 2.2 VBK süreci (Fayyad vd., 1996)

2.2. Bilgi Keşif Süreci Tanıtımı

Şekil 2.2, VBK sürecinde yer alan adımları göstermektedir. *Veri seçimi*, birkaç veri kümesinin birleştirilerek, seçilen probleme uygun örneklem kümesinin elde edilmesidir.

Önişleme veya temizleme; seçilen örneklemde yer alan ilgisiz niteliklerin atıldığı, tekrarlı kayıtların ayıklandığı, hatalı tutanakların çıkarıldığı ve Boş nitelik değerlerinin çıkarıldığı veya değiştirildiği aşamadır. Bu aşama keşfedilen bilginin kalitesini arttırır. *Veri dönüştürme*, seçilen örneklemde bulunan tutarsızlıkların giderildiği ve var olan niteliklerden faydalanılarak yeni niteliklerin elde edildiği adımdır. Bu aşama VM'de oluşturulan modelinin geçerliliğini artırır. *Veri madenciliği veya model oluşturma*, seçilen bir VM algoritmasının (sınıflandırma, kümeleme, eşleştirme, vb.) işletilmesidir. *Değerlendirme veya denetleme*, keşfedilen bilginin geçerlilik, yenilik, yararlılık ve basitlik kıstaslarına göre değerlendirilmesi aşamasıdır.

VBK sürecinde izlenmesi gereken temel aşamalar ise şunlardır:

2.2.1. Problemi tanımlama

Model oluşturulmadan önce üzerinde çalışılan veri iyi bir şekilde anlaşılmalı ve çözülmeye çalışılan iş problemi açıkça tanımlanmalıdır. Bu işlem; iş gereksinimlerini analiz etme, problemin kapsamını tanımlama, modelin değerlendirileceği ölçüm biçimini tanımlama ve VM projesi için en son hedefi tanımlamayı içerir. Bu görevler aşağıdaki sorulara dönüştürülebilir:

- Veri kümesinde hangi nitelik tahmin edilmeye çalışılmaktadır?
- Ne tür ilişkiler bulunmaya çalışılmaktadır?
- VM modeli ile tahmin mi yapılacak yoksa sadece ilginç desen ve birliktelikleri ortaya çıkarmak mı istenmektedir?
- Verideki dağılım nasıldır?
- Sütunlar nasıl ilişkili ve eğer çok sayıda tablo ile çalışılıyor ise bu tablolar nasıl ilişkilidir?
- Eldeki veri problemin gerektirdiği analiz türlerini desteklemekte midir? veya problemin çözümünde iç veya dıştan ek veriler mi gerekmektedir?
- İş problemini hangi tür analizler çözebilir?

Bunlar, veriler üzerinde çalışmaya başlamadan önce cevap verilmesi gereken sorulardır. Cevapları bulabilmek için veri uygunluğu çalışması ardından iş kullanıcılarının ihtiyaçlarını uygun veriye göre keşfetmek gerekir. Eğer veriler

kullanıcıların ihtiyaçlarını ortaya çıkarmakta destek vermezse, projede tekrar tanımlamaya ihtiyaç duyulur.

2.2.2. Veri hazırlama

Problem tanımlandıktan sonra çözüme giderken ilk olarak çalışılan iş problemi ile ilgili ham verilerin bulunması gerekmektedir. Veriyi toplamak hantal bir iştir. Genellikle veriler, şirket veya işletme çapında dağılmış ve farklı formatlarda saklanmaktadır. Toplanan bu verilerin modelde kullanılabilecek hale getirilinceye kadar gerçekleştirilen işlemlerin tümü veriyi hazırlama kapsamındadır.

2.2.2.1. Veri temizleme

Veri temizleme; veriyi inceleme, gereksiz sütunları çıkarma, ve kalan sütunlarda var ise hatalı veya tutarsız değerleri düzeltme olarak özetlenebilir. Veritabanına bilgi girişi sırasında bazı hatalı girişler yapılmış olabilir. Dikkatli olunmaz ise bu problemler, modelin etkinliğini belirgin şekilde düşürebilir.

Veri kümesinde bir çok çeşit tutarsızlıklar olabilir. Aşağıda bunlardan bazıları verilmektedir:

- Çok sayıda boş değer (null) içeren sütunlar,
- Çok az veya çok fazla farklı değer içeren sütunlar,
- Telefon numaraları gibi her kayıta bire bir ilişkili sütunlar veya tek duruma sahip sütunlar,
- Sütunun normal dağılımının çok fazla dışında kalan kayıtlar,
- Gerçek hayata uymayan değerler (Negatif değerde aylık gelir veya bir ürünün satın alınma tarihinin, satın alanın doğum tarihinden önce olması gibi.),
- Belirli formata uymayan kayıtlar,
- Farklı tarih formatları,
- Farklı niteliklere sahip benzer kayıtlar,

Bu ve benzeri problemleri çözmenin yolu; modelin gereksinimleri ve seçilen problem yaklaşımına göre değişir. Örneğin hücrelerdeki sınır dışı değerler; sütunun

ortalama değeri veya belirli bir dağılım türüne uyan değer ile değiştirilebilir veya sınır dışı değer taşıyan hücre, satırın geri kalanı ile birlikte, çıkarılabilir.

Genelde çok geniş veri kümesi ile çalışılırken her bir muameleye bireysel olarak bakılamamaktadır. Bu nedenle veriyi inceleme ve tutarsızlıkları bulmak ve düzenlemek için bazı otomasyon yöntemlerinden faydalanılır. Örneğin sütun bazında en küçük, en büyük, ortalama ve standart sapma değerleri gibi çeşitli değerler hesaplanarak, veri dağılımına bakılabilir ve sonuçta hangi veriler tutarsız görünüyorsa, bunlar belirlenen bir strateji ile giderilmeye çalışılır.

2.2.2.2. Veri dönüştürme

Veri hazırlamada çoğu zaman VM modelini oluşturmadan önce veri kümesinin bazı sütunlarını dönüştürmek gerekmektedir. Model oluşturmak için kullanılan veriler bir çalışanın aylık bilgisi gibi çok sayıda olası değerleri içerebilir. Sütunlar, çok sayıda durumları ile süreklilik gösteren bir karakterdedir. Modelin daha fazla anlamlı sonuçlar ortaya çıkarabilmesi için veri kesikli hale getirilerek düşük, orta, yüksek gibi sınırlı sayıda değer aralıkları oluşturulabilir.

Yine mevcut sütunlara dayalı olarak yeni bir sütun tanımlamak istenebilir. Örnek olarak bir çalışanın sağlık sigortası gibi tüm ücret kesintilerinin toplamını içeren bir sütun olmayabilir; ancak her bir maliyet değeri toplanıp yeni bir sütunda gösterilerek bu durum başarılabilir.

2.2.2.4. Veri inceleme

VM modeli hazırlarken, modelin etkinliğine yardımcı olan veya engel olan sütunlara karar vermek gerekmektedir. Bu nedenle verilerin sütun boyunca nasıl dağıldığı ve farklı sütunların bir diğeri ile veya belirlenmiş ise hedef sütun ile nasıl ilişkide olduğuna bakılır. İşte bu veriyi inceleme sürecidir.

Veri incelemeye gelindiği zaman, görsel ve sayısal teknikler verilerin nasıl etkileşimde olduğu ve bağımsız değişkenlerin seçilmesi hakkında tek bir bakış açısı sunarlar. Görsel ve sayısal teknikler birlikte veri hakkında daha derin bir anlayış sağladığından VM araçlarında genellikle bu iki teknik birlikte kullanılmaktadır.

Görsel teknikler, çok fazla sayıda sütuna hızlı bir şekilde bakma ve bunların arasındaki etkileşim hakkında genel bir fikir verirler. Karakter tabanlı (varchar) veriyi görselleştirmek için histogramlar, sayısal veriyi göstermek için hem histogram hem de noktasal grafikler kullanılabilir.

Histogramlar, sütunların farklı durumlarının hedef sütuna göre dağılımını tanımlar. Histogramı oluşturabilmek için az sayıda durum ile hedef sütun karşılaştırılabilir. Bunun anlamı özellikle sayısal sütunlarda veriyi gruplamak gerekir. Aylık ücret değeri örneğin 10.000 farklı durum içermesine rağmen bu veri yüksek, orta ve düşük gibi üç yeni duruma dönüştürülebilir. Bu hali ile histogram oluşturularak bir sütuna ait farklı durumların hedef sütuna olan etkisi incelenebilir. Karakter tabanlı sütunlar için genellikle böyle bir gruplamaya gerek kalmaz; doğrudan sütun üzerindeki veri ile çalışılabilir.

Noktasal grafik, hedef sütunların durumlarının seçili sütuna karşı nasıl dağıldığını ifade eder. Burada X-ekseni giriş sütununu, Y-ekseni ise hedef sütunu tutar. Hedef sütunun durumları, seçili sütuna karşı yansız bir şekilde yayılıyor ise, seçili sütunun hedef sütun hakkında anlamlı bir şey ifade etmediği; hedef sütunun durumları gruplaşmış durumda ise seçili sütunun hedef sütunun sonucuna karar verirken kullanışlı olduğu yargısı verilebilmektedir.

Sayısal teknikler, verinin nasıl etkileşimde olduğu hakkında daha fazla somut anlayış verirler. "Sayısal" teriminin ima ettiği gibi, sayısal teknikler yalnızca sayısal veri içeren sütunlarda uygulanabilirler.

Sayısal bir inceleme olan korelasyon matrisi kullanılarak, sütunların aralarındaki ilişki veya her bir sütunun hedef sütun ile arasındaki ilişki incelenebilir. Korelasyon, bir sütunun bir diğer sütunla karşılaştırılarak nasıl değiştiğini tanımlar. Hesaplanan korelasyon değeri, -1 ila 1 arasında bulunur. Eğer bir sütun değerlerindeki artış, ikinci bir sütunun değerlerindeki artışa uymakta ise korelasyon pozitif; bir sütun değerlerindeki artış, ikinci sütun değerlerindeki düşüşe uymakta ise korelasyon negatiftir. Korelasyonun hesaplanan değeri, değişkenler arasındaki ilişkinin kuvvetine göre, -1 ve 1 e yaklaşır. İdeal korelasyon, 1 ve -1 değerini alır.

2.2.3. Model oluşturma

Model oluşturmada aşağıdaki işlevler gerçekleştirilir:

- Sütunların seçimi,
- Modelin seçimi,
- Parametreleri ayarlama,
- Modeli eğitme.

İlk olarak, kurulacak modele bağlı veri seçimi yapılır. Örneğin tahminleyici bir model için, bu adım bağımlı ve bağımsız değişkenlerin ve modelin eğitiminde kullanılacak veri kümesinin seçilmesi anlamını taşımaktadır.

Model oluşturma esnasında veri kümesindeki sütunların tümünü kullanmak; kaynak harcayan, zaman alıcı ve anlamayı zorlaştıran bir durumdur. Modele eklemek için ilgili olduğu doğrulanmış sütunları seçmek daha iyidir. Sıra numarası, kimlik numarası gibi anlamlı olmayan ve diğer değişkenlerin modeldeki ağırlığının azalmasına da neden olabilecek değişkenlerin modele girmemesi gerekmektedir.

Model oluşturulmadan önce, asıl veri kümesi, eğitim ve test veri kümeleri olarak rasgele ikiye ayrılmalıdır. Eğitim veri kümesi, modeli oluşturmakta kullanılır. Verideki gizli desenleri öğrenerek hedeflenen amaca ulaşmak için VM algoritmaları kullanılır. Ardından modelin doğruluğunu denetlemek için test veri kümesi üzerinde tahmin edici sorgular oluşturulur. Test verisi modeli eğitmek için kullanılan veri ile aynı veri kümesinden geldiği için modelin performansının doğruluğu hesaplanabilir. Genelde çalışılan tablo veya veri kümesi tek olduğundan eğitim ve test tablolarını bu asıl tablodan yapay olarak oluşturmak gerekmektedir.

Tabloyu ayırmakta hedef, asıl tabloyu her biri doğru bir şekilde asıl tabloyu temsil eden iki tabloya bölmektir. Bu hedefe ulaşmak için aşağıda belirtilen durumların sağlanması gerekmektedir (Paul vd., 2004):

- Ayrılan tabloların her ikisi de aynı yapıda olmalıdır. -iki veri kümesi de aynı sütunlara aynı isimle sahip olmalıdır.-
- Eğitim tablosunda bulunan satırlar, test tablosunda bulunmamalıdır. -satırlar her bir tablo için benzersiz olmalıdır.-

- İyi bir model oluşturabilmek ve yine iyi bir denetleme gerçekleştirebilmek için her iki tablo da yeterli sayıda kayıt (satır) bulundurmalıdır.
- Giriş ve hedef sütunlardaki verilerin dağılımı iki tabloda da yaklaşık olarak benzemelidir.

Veriler, örneklem boyunca rasgele dağılmış olmayabilir. Bu nedenle tabloların yansız bir şekilde ayrılmasını sağlamak için asıl tablodan her bir tablo için belirtilen yüzdelerde farklı satırlar rasgele seçilir.

Modeli eğitmek için veriler belirli algoritmalardan geçirilir. Her algoritma modelin sonucunu etkileyebilen ayarlanabilir parametreler içerir. Modele dahil edilen her bir sütun bir kaç şekilde parametrelendirilebilir. Bu durum modelin nasıl oluşturulduğu konusunda derin etkiye sahiptir. Sütunlar, modele eklenirken sütunların hakkında bazı bilgiler gereklidir. Anahtar, giriş veya hedef sütunu olduğu veya sütun içerisindeki dağılımın sürekli veya kesikli olması gibi.

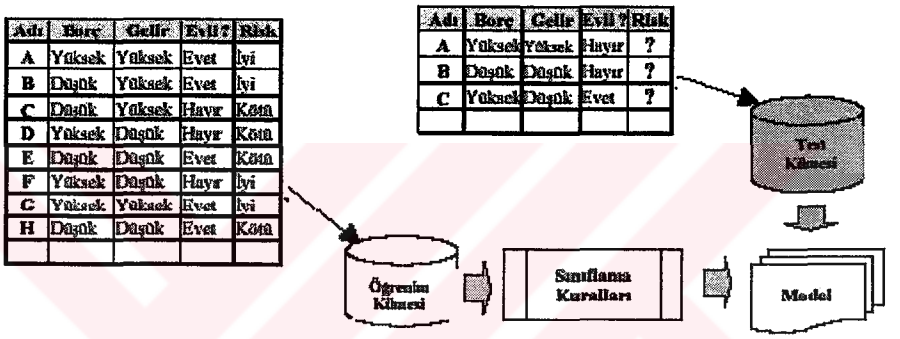
Aşağıda sütunları tanımlamak için kullanılan bazı özellikler bulunmaktadır:

- Veri tipi,
- Kullanım,
- İlgili,
- Dağılım,
- İçerik tipi,
- İşaret modelleme.

Eğitim işleminin sonucunda oluşan matematiksel model, kümeleme algoritmasında veriyi incelemek için, karar ağacı algoritmasında ise tahmin oluşturmak için kullanılabilir. Modele dahil edilen sütunlar ne kadar iyi seçildiği ve sonuçta modelin parametreleri nasıl değiştirildiğine göre modelin performansına karar verilir.

Tanımlanan problem için en uygun modelin bulunabilmesi, olabildiğince çok sayıda modelin kurularak denenmesi ile mümkündür. Bu nedenle veri hazırlama ve model kurma aşamaları, en iyi olduğu düşünülen modele varılıncaya kadar yinelenen bir süreçtir.

Model kuruluş süreci, denetimli ve denetimsiz öğrenimin[1] kullanıldığı modellere göre farklılık göstermektedir. Örnekten öğrenme olarak da isimlendirilen denetimli öğrenimde, bir denetçi tarafından ilgili sınıflar önceden belirlenen bir kritere göre ayrılarak, her sınıf için çeşitli örnekler verilir. Sistemin amacı, verilen örneklerden hareket ederek her bir sınıfa ilişkin özelliklerin bulunması ve bu özelliklerin kural cümleleri ile ifade edilmesidir. Öğrenme süreci tamamlandığında, tanımlanan kural cümleleri verilen yeni örneklere uygulanır ve yeni örneklerin hangi sınıfa ait olduğu kurulan model tarafından belirlenir.



Şekil 2.3 Denetimli öğrenme (Akpınar, 2000)

Denetimli öğrenimde seçilen algoritmaya uygun olarak ilgili veriler hazırlandıktan sonra, ilk aşamada verinin bir kısmı modelin öğrenimi, diğer kısmı ise modelin geçerliliğinin test edilmesi için ayrılır. Modelin öğrenimi öğrenim kümesi kullanılarak gerçekleştirildikten sonra, test kümesi ile modelin doğruluk derecesi belirlenir.

Denetimsiz öğrenmede, kümeleme analizinde olduğu gibi ilgili örneklerin gözlenmesi ve bu örneklerin özellikleri arasındaki benzerliklerden hareket ederek sınıfların tanımlanması amaçlanmaktadır.

---

[1] Supervise and Unsupervised Learning

2.2.4 Model geçerliliği

Model oluşturulduktan sonra performansının ne kadar iyi olduğunu bilmek gerekmektedir. Bu nedenle modelin etkinliğini denetleyen bir ölçü tanımlanmalıdır. Çoğunlukla birden fazla model üretilir ve birbirleri arasında performans karşılaştırılması yapılır.

Bir modelin doğruluğunun test edilmesinde kullanılan en basit yöntem *geçerlilik testi*dir. Bu yöntemde tipik olarak verilerin % 5 ile % 33 arasındaki bir kısmı test verileri olarak ayrılır ve kalan kısım üzerinde modelin öğrenimi gerçekleştirildikten sonra, bu veriler üzerinde test işlemi yapılır. Bir sınıflandırma modelinde yanlış olarak sınıflanan kayıt sayısının, tüm kayıt sayısına bölünmesi ile hata oranı, doğru olarak sınıflanan kayıt sayısının tüm kayıt sayısına bölünmesi ile ise doğruluk oranı hesaplanır. Dolayısıyla doğruluk ve hata oranı toplamı daima 1'dir. (Doğruluk Oranı = 1 - Hata Oranı)

Sınırlı miktarda veriye sahip olunması durumunda, kullanılabilecek diğer bir yöntem çapraz geçerlilik testidir. Bu yöntemde veri kümesi tesadüfi olarak iki eşit parçaya ayrılır. İlk aşamada A parçası üzerinde model eğitimi ve B parçası üzerinde test işlemi; ikinci aşamada ise B parçası üzerinde model eğitimi ve A parçası üzerinde test işlemi yapılarak elde edilen hata oranlarının ortalaması kullanılır. Bir kaç bin veya daha az satırdan meydana gelen küçük veritabanlarında, verilerin n gruba ayrıldığı n katlı çapraz geçerlilik testi tercih edilebilir. Verilerin örneğin 10 gruba ayrıldığı bu yöntemde, ilk aşamada birinci grup test, diğer gruplar öğrenim için kullanılır. Bu süreç her defasında bir grubun test, diğer grupların öğrenim amaçlı kullanılması ile sürdürülür. Sonuçta elde edilen 10 hata oranının ortalaması, kurulan modelin tahmini hata oranı olacaktır.

Küçük veri kümeleri için modelin hata düzeyinin tahmininde kullanılan bir başka teknik önyüklemedir (bootstrapping). Çapraz geçerlilikte olduğu gibi model bütün veri kümesi üzerine kurulur. Daha sonra en az 200, bazen binin üzerinde olmak üzere çok fazla sayıda öğrenim kümesi tekrarlı örneklemelerle veri kümesinden oluşturularak hata oranı hesaplanır.

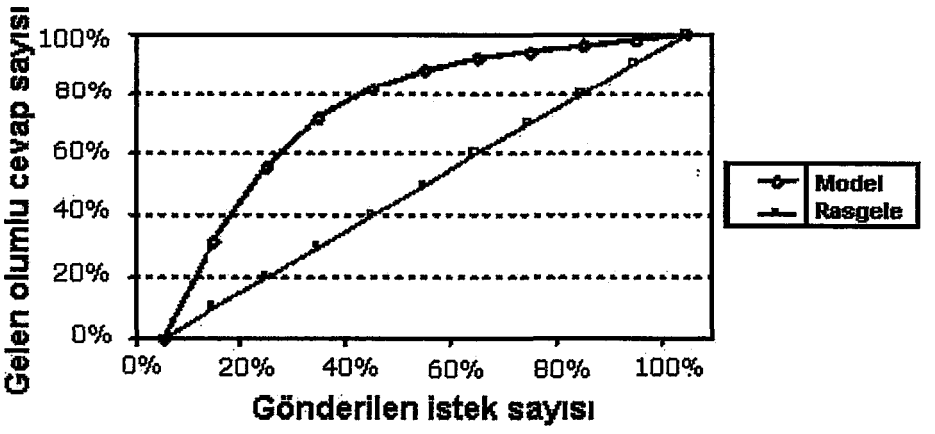
Özellikle sınıflandırma problemleri için kurulan modellerin doğruluk derecelerinin değerlendirilmesinde basit ancak faydalı bir araç olarak risk matrisi kullanılmaktadır.

Aşağıda bir örneği görülen bu matriste sütunlarda fiili, satırlarda ise tahmini sınıflandırma değerleri yer almaktadır. Örneğin, fiilen B sınıfına ait olması gereken 46 elemanın, kurulan model tarafından 2'sinin A, 38'inin B, 6'sının ise C olarak sınıflandırıldığı Tablo 2.1'deki matriste görülebilmektedir.

Tablo 2.1 Risk matrisine bir örnek

	Fiili		
Tahmini	*A Sınıfı*	*B Sınıfı*	*C Sınıfı*
A Sınıfı	45	2	3
B Sınıfı	10	38	2
C Sınıfı	4	6	40

Kaldıraç oranı veya grafiği bir modelin sağladığı faydanın değerlendirilmesinde kullanılan önemli bir yardımcıdır. Tahminleyici model kullanılarak ve tahminleyici model kullanılmadan gerçekleştirilen sonuçları karşılaştırarak modelin etkinliğini ölçer. Bunu gerçekleştirebilmek için test veri kümesindeki bir sütunu tahmin eder, sonra bunu gerçek değerler ile karşılaştırır. Daha sonra tahmini değerler ve gerçek değerler grafiksel olarak görüntülenir. Şekil 2.4'de, bir şirketin rasgele seçerek gönderdiği müşteri istek postalarına nazaran modelin seçtiği müşterilerin hep daha yüksek oranda olumlu cevap verdiği dikkat çekicidir.



Şekil 2.4 Kaldıraç grafiği

Model tarafından önerilen uygulamadan elde edilecek kazancın bu uygulamanın gerçekleştirilmesi durumunda katlanılacak maliyete bölünmesi ile edilecek oran da modelin geçerlilik kriteri olarak kullanılabilmektedir.

Modelin geçerliliği için önemli bir kriter modelin anlaşılabilirliğidir. Bazı uygulamalarda doğruluk oranlarındaki küçük artışlar çok önemli olsa da, bir çok işletme uygulamasında ilgili kararın niçin verildiğinin yorumlanabilmesi çok daha büyük önem taşıyabilir. Çok ender olarak yorumlanamayacak kadar karmaşıklaşsalar da, genel olarak karar ağacı ve kural temelli sistemler model tahmininin altında yatan nedenleri çok iyi ortaya koyabilmektedir.

Kurulan modelin doğruluk derecesi ne derece yüksek olursa olsun, gerçek dünyayı tam anlamı ile modellediğini garanti edebilmek mümkün değildir. Yapılan testler sonucunda geçerli bir modelin doğru olmamasındaki başlıca nedenler, model kuruluşunda kabul edilen varsayımlar ve modelde kullanılan verilerin tam doğru olmamasıdır. Örneğin modelin kurulması sırasında varsayılan enflasyon oranının zaman içerisinde değişmesi, bireyin satın alma davranışını belirgin biçimde etkileyecektir.

2.2.5. Modelin kullanılması, izlenmesi ve veri yönetimi

Model oluşturulduktan ve geçerliliği ölçüldükten sonra üretim ortamına yerleştirilir. Burası modelin iş karar verme biriminde kullanıldığı ve buraya kadar olan bütün çalışmaların sonuçlarının gözükmeye başladığı yerdir. Kurulan modeller, risk analizi, kredi değerlendirme, dolandırıcılık tespiti gibi işletme uygulamalarında doğrudan kullanılabileceği gibi, promosyon planlaması gibi bir başka uygulamanın içine gömülebilir.

Modelin uygulandığı kurum veya organizasyona daha fazla veri geldikçe modeli güncellemeye ihtiyaç duyulur ve bu da sonuç olarak modelin geçerliliğini arttıracak bir etki yapar. Zaman içerisinde bütün sistemlerin özelliklerinde ve dolayısıyla ürettikleri verilerde ortaya çıkan değişiklikler, kurulan modellerin sürekli olarak izlenmesini ve gerekiyorsa yeniden düzenlenmesini gerektirecektir. Hedef ve giriş değişkenlerinde gerçekleşen farklılıkları gösteren grafikler, modelin izlenmesinde kullanılan yararlı bir yöntemdir.

Veriyi araştırma ve modeli oluşturma ile ilgili bilgiler, kaydedilmesi gereken kullanışlı bilgilerdir. Bu nedenle modelden çıkarılan sütunlar, daha önceden oluşturulmuş modeller ve bu modellerin geçerlilikleri veritabanı veya benzeri bir kayıt ortamında tutulur. Bu durum, araştırmada belirli tablo veya sütunların istendiğinde çıkarabilme veya dahil edebilme veya zaman tasarrufu sağlamak için çalışılan tablonun daha az kayıt taşıyan bir örneklemi ile denemeler yapabilme gibi esneklikler kazandırır.

2.3. Veri Madenciliği

VM yeni bir teknoloji olmasına karşın hızlı bir gelişme kaydetmektedir. Kullanılan alanları, kullandığı teknikler ve araçlar yönünden çok fazla çeşitlilik göstermesi nedeniyle VM'nin tanımı konusunda da literatüre bakıldığında çok geniş bir çeşitlilik gözlenmektedir. En çok kabul gören tanımlarından biri; "VM, büyük hacimli veritabanlarından geçerli, yeni, yararlı ve anlaşılabilir veri desenlerinin ortaya çıkarılmasıdır" (Frawley vd., 1991) şeklindedir. Aşağıda literatürde öne çıkan diğer tanımlar verilmiştir:

"VM, veritabanları ve/veya veri ambarları[2] gibi veri kaynaklarında depolanan büyük miktarlardaki verilerden şablonlar, birliktelikler, değişiklikler, anormallikler ve önemli yapılar gibi ilginç bilgileri keşfetme işlemidir." (Fayyad vd., 1996)

"VM; veri ambarlarında tutulan çok çeşitli verilere dayanarak daha önce keşfedilmemiş bilgileri ortaya çıkarmak, bunları karar vermek ve eylem planını gerçekleştirmek için kullanma sürecidir." (Swift, 2001)

"VM, örüntü tanıma teknolojilerini, istatistiksel ve matematiksel teknikleri kullanarak büyük hacimlerdeki verilerden anlamlı yeni korelasyonlar, kalıplar ve trendler çıkarma sürecidir." (http://www.spss.com/)

"VM, eldeki verilerden üstü kapalı, çok net olmayan, önceden bilinemeyen; ancak potansiyel olarak kullanışlı bilgilerin çıkarılmasıdır. Başka bir deyişle, VM, verilerin içerisindeki desenlerin, ilişkilerin, değişimlerin, düzensizliklerin, kuralların ve

---

[2] Datawarehauses

istatistiksel olarak önemli olan yapıların yarı otomatik olarak keşfedilmesidir." (Vahaplar vd, 2002)

"Büyük miktarda veri içinden gelecekle ilgili tahmin yapmamızı sağlayacak bağıntı ve kuralların aranmasıdır." (Bontempo vd., 1995)

VM tanımları incelendiğinde, bu tanımlarda bulunan ortak olan unsurlardan ilki "büyük" miktarlarda verinin veritabanlarında tutulması, ikincisi ise bu verilerden "anlamlı" bilgiler elde edilmesidir. VM'de kullanılan veritabanları hakkında sıkça bahsi geçen *genişlik veya yüksek hacim* tek bir iş istasyonunun belleğine sığamayacak kadar büyük veri kümelerini; *dağıtık* yapı ise, farklı coğrafi konumlarda bulunan verileri bulundurabilmeyi ifade etmektedir.

Temel olarak VM, veri kümeleri arasındaki örüntülerin ya da düzenliliklerin, verinin analizi ve yazılım tekniklerinin kullanılarak bulunması ile ilgilidir. Veriler arasındaki ilişkiyi, kuralları ve özellikleri belirlemekten bilgisayar sorumludur.

VM tekniklerini uygulayarak işletmeler, müşterilerin satın alma örüntü ve davranışları hakkında veriden tamamen istifade edebilir ve bu şekilde müşteri motivasyonunu daha fazla anlama imkanı elde edebilir. Bu durum; VM tarafından sunulan otomatikleştirilmiş analizleri, tipik olarak karar destek sistemlerinde[3] olduğu gibi manüel araçlar tarafından sağlanan geçmiş olayların analizinin ötesine taşır. VM araçları geleneksel olarak çözümü çok zaman alıcı olan iş sorunlarına cevap verebilmektedir.

2.3.1. VM tarihsel gelişimi ve diğer teknolojiler ile etkileşimi

İş verilerinin iş bilgisi haline dönüşmesi, 1960'lı yıllarda veri toplama[4] sistemlerindeki gelişme ile başlar. Veri toplama, geçmişle ilgili soruları cevaplar. 1980'li yıllara gelindiğinde ilişkisel veritabanları[5] sayesinde veri erişim teknikleri ile tanışıldı. 1990'larda çok boyutlu veritabanları[6], OLAP[7] ile birlikte VA (Veri Ambarlama)[8] ve

---

[3] Decision support system
[4] Data collection
[5] Relational databases
[6] Multidimensional databeses

karar destek sistemleri bulundu. Günümüzde VM; kullandığı gelişmiş algoritmalar, çok işlemcili bilgisayarlar ve geniş veritabanları ile yeni bir kavramdır. VM'yi önceki kavramlardan ayıran temel fark, VM'nin geleceğe ait sorulara cevap vermesindedir (Bontempo vd., 1995).

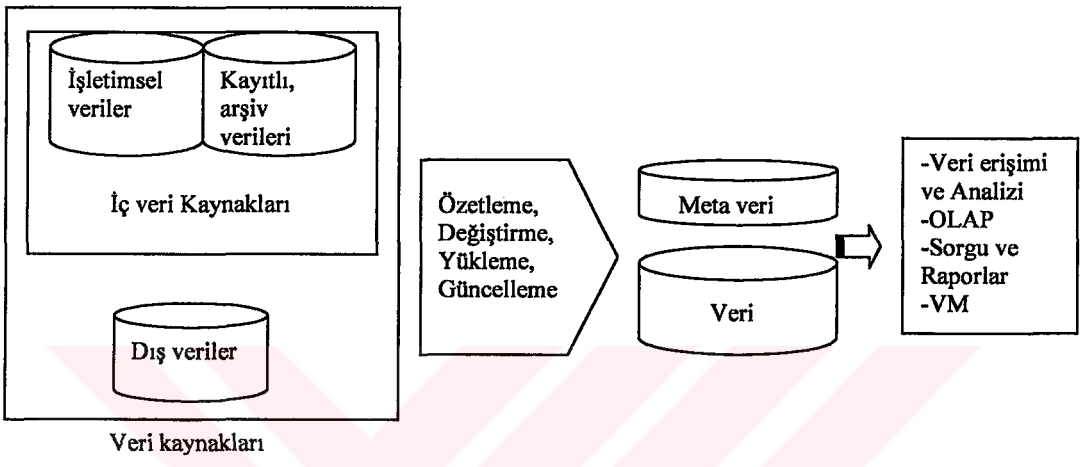
VM, büyük miktarda veriyi inceleme amacı üzerine kurulmuş olduğu için veritabanı teknolojileri ile yakından ilişkilidir. Verilerin amaca uygun bir şekilde saklanması ve gerek duyulduğunda da hızlı bir şekilde ulaşılabilmesi gerekmektedir. Günümüzde bu amaçla veri ambarları, yaygın olarak kullanılmaya başlanmıştır. Veri ambarı, mesleki yarar sağlamak için sorgulamaya açık merkezi bir veri havuzu olarak tanımlanabilir. Paylaşılabilir veri kaynaklarını depolamak için karar destek sistemleri için bir veritabanı uygulamasıdır ve günlük kullanılan veritabanlarının birleştirilmiş ve işletilmeye daha uygun bir özetini saklamayı amaçlar. Veri ambarlarının anahtar fonksiyonları; arşivlenmiş işlevsel veriyi seçip çıkarmak, farklı veri formatlarının arasındaki tutarsızlıkların üstesinden gelmek, şirkete ait değişik yerlerdeki veriyi entegre etmek ve eklenen bilgileri birleştirmektir. Veri ambarları, genelde özelleştirilmiş geniş bir şirketten ziyade basit bir bölümdeki durumları izleyen karar destek veritabanları topluluğundan oluşur (Berry vd., 2000, Berson vd., 1997, Data Mining '99: Technology Report, 1999).

Veri ambarlarında verileri tutmak ve VM uygulamalarında kullanmak, geleneksel işleri elektronik ortama taşıma ve elektronik işletme olmanın da gereklerinden ve ileri aşamalarından biridir. Elektronik işletme olabilmek için satış ve kar artırıcı amaca yönelik uygulamalara yönelmek gerekmektedir. Bu da, ancak müşteri odaklı çalışmakla[9], müşterilerle ilgili gerekli verileri tutarak anlamlı bilgiler elde etmekle dolayısıyla çok sayıda müşteriyle bire bir ilişki kuracak teknolojik ve yönetimsel altyapının kurulmasıyla mümkün olabilmektedir. Bu nedenle işletmeler, VM'yi kullanılarak veri ambarlarından anlamlı eğilimleri ve desenleri seçip çıkartmakta faydalanırlar (Özmen, 2002).

---

7 Çevrim-içi analitik işleme
8 Datawarehousing
[9] Customer Relationship Management

Şekil 2.5'te görüldüğü gibi iç ve dış veri kaynaklarındaki verilerden istenen özet veriler seçilerek ve gerekli ön işlemeden sonra veri ambarında standart bir formda saklanır. Ardından veri amaç doğrultusunda veri ambardan alınarak VM ve diğer araştırma tekniklerinin çalışmasında kullanılır.



Şekil 2.5 Veri ambarı yapısı

Veri ambarında veri oluşturulduktan sonra bu verinin elle veya gözle analizi yapılabilir. Bunun için OLAP[10] (Online Analytical Processing) programları kullanılır. Bu programlar, veriye her boyutu veride bir alana karşılık gelen çok boyutlu bir küp olarak bakmayı ve incelemeyi sağlar. Böylece, boyut bazında gruplama, boyutlar arasındaki korelasyonları inceleme ve sonuçları grafik veya rapor olarak sunma olanağı sağlar. Bu amaçla SQL[11] (Structured Query Language) kullanılır. OLAP ile kullanıcı yönelimli olarak hipotez üretilir ve bu hipotezlerin denetimi yine OLAP araçları ile gerçekleştirilir. VM araçları ise veri üzerinden hipotez üretmede kullanılır.

OLAP araçları, işletme çapında veritabanı sistemleri ile bağlı haldedir ve analiz eden kişiler ve yöneticilere iş performansını izleme izni verir. VM teknikleri, OLAP

---

[10] Çevrim içi analitik işleme
[11] Yapısal sorgulama dili

tarafından görüntülenmiş ve özetlemiş veri üzerinde daha derin ve çok boyutlu bilgi sağlaması için uygulanabilir. Bununla birlikte, VM teknikleri için genelde OLAP veya OLAP ın analitik uzantısından farklı amaçları sağlayan yaklaşımı temsil ettiği şeklinde düşünülür. VM, geleceği tahmin etmek için model oluşturmaya yardım edebilirken OLAP araçları ise sadece tarihsel eğilimleri raporlar (Berson vd., 1997, Berry vd., 2000).

VM aynı zamanda veritabanı teknolojisi, istatistik, YZ (Yapay Zeka)[12], MÖ(Makine Öğrenimi)[13], örüntü tanıma[14] ve veri görselleştirme[15] gibi birçok teknik alanı bir araya getiren çok disiplinli bir alandır (Fayyad vd., 1996).

İstatistik, veriden anlam çıkarma probleminin kalbinde yer alır. Tüm verinin belirli bir örneğinden genel desenleri çıkarmak istendiğinde çıkan belirsizliği ölçmede ortak bir dil ve çatı sağlar (Fayyad vd., 1996). Hem hipotez geçerliliği ve hem de veri analizi araştırmalarında istatistiksel teknikler, temel oluşturan önemdedir. Glymour vd. (1996), istatistiksel bakışı vermektedir. Klasik istatistik kavramları, VM'de izlenilen yol hakkında anlama kabiliyetini artırmada ve yön göstermektedir. Bunlar; VM'nin içerisinde yer alan araştırma mimarisinin gerektirdiği hipotez değerlendirme, araştırma sonucunu değerlendirme ve sonuçların uygun kullanımını ifade eder. İstatistik, belki araştırma mimarilerini anlamada çok az katkı sunabilir; ancak araştırmanın hipotez değerlendirmesinde, araştırma sonuçlarının değerlendirilmesinde ve sonuçların uygulanmasında ortaya çok değerli anlamlar sunarlar. Bu noktada istatistik, VM süreci içerisinde kendi başına bir çözüm değil; çözüme ulaşmak için verilecek karar sürecini destekleyen, problemi çözmek için gerekli olan bilgileri sağlamaya yarayan bir araçtır. Aslında VM'de vurgulanan unsurlar ile istatistiğin uygulama amaçları yakınlık göstermektedir. Her ikisinde de temel olan öğeler *veri* ve *bilgi*dir. Yani verilerden bilgiyi keşfetmek. Bu birliktelik sayesinde istatistik bilimi ile uğraşan kişiler "VM, istatistik

---

[12] Artificial intelligence
[13] Machine learning
[14] Image processing
[15] Data visualization

biliminin teknolojiyle bütünleşmesi sonucu oluşturulan bir araçtır" diyebilmektedirler (Özmen, 2002).

VM ile MÖ arasındaki yakın bağ da kolaylıkla görülebilir. Bu iki disiplin veri içerisindeki ilginç düzenlilikleri ve örüntüleri bulmayı amaçlar. MÖ, öğrenme işleminin otomasyonudur. Öğrenme, çevresel durum ve geçişlerin gözlemlerine göre kurallar bütünüdür. MÖ; geçmiş örnekleri ve sonuçlarını inceler, bunları nasıl ürettiğini öğrenir ve yeni durumlar hakkında genelleştirmeler yapar (Mitchell, 1999). MÖ yöntemleri VM algoritmalarında kullanılan yöntemlerin çekirdeğini oluşturur. MÖ'de kullanılan karar ağacı[16], kural tümevarımı pek çok VM algoritmasında kullanılmaktadır. MÖ ile VM arasında benzerliklerin yanı sıra farklılıklar da göze çarpmaktadır. Öncelikle VM algoritmalarında kullanılan örneklem boyutu, MÖ'de kullanılana nazaran çok büyüktür. Genellikle MÖ'de kullanılan örneklem boyutu 100 ile 1000 arasında değişir. VM uygulaması, kontrol edilemeyen gerçek hayat verisini ele alabilecek biçimde MÖ tekniklerini genişletir (Oğuz, 2000). VM algoritmaları milyonlarca gerçek hayat nesneleri üzerinde uğraşmaktadır ki; bunların karakteristiği boş, artık, gürültülü değerler olarak belirlenebilir. Bu nedenle VM algoritmaları, aynı zamanda bilgi keşfetmeye uygun nesne niteliklerinin elde edilme sürecindeki karmaşıklıkla da baş etmek zorundadır.

VM'de amaç, iş süreçlerinde karar vericilerin üzerinde hareket edeceği yeni bilgiler üretmek ve bunu yaparken de kullanıcının bilgi çıkarma sürecinde katkısının olabildiğince az tutulması, işin olabildiğince otomatik olarak yapılabilmesidir. Bu işlemi; şirket işlemleri, müşteri geçmişi ve demografik bilgilerini içeren çeşitli veri kaynakları ve kredi büroları gibi harici kaynaklardan toplanan veriye dayalı gerçek dünyanın bir modelini oluşturmak için YZ gibi teknikler kullanarak yapar (Akpınar, 2000). Bu model, verideki desenleri üretir. Üretilen bu bilgi, karar vermeye ve yeni iş imkanlarına destek verir.

---

[16] Decision tree

2.3.2. VM kullanım alanları

VM uygulamaları, endüstriden ticarete birçok alanda boy gösterir. Örnek olarak iletişim, stok giriş–çıkışı, kredi kartı ve sigorta kuruluşları; VM'yi kendi servislerinde dolandırıcılığa karşı kullanmaktadırlar. Tıp endüstrisi, VM'yi tıbbi testlerin etkinliğini tahmin etmek için kullanır. Perakendecilik sektöründe VM, kampanya ve reklam faaliyetlerinin etkinliğinin değerlendirilmesinde kullanılmaktadır. Aşağıda çeşitli endüstri ve servis sektörlerindeki ilgili uygulamalar listelenmiştir (Brachman vd., 1996; Fayyad vd., 1996; Data Mining '99: Technology Report, 1999; Mitchell, 1999; Berry vd., 2000):

Üretim

- Parça hata tanısı,
- İşlem modelleme,
- Kalite kontrol,
- Kaynak atama.

Pazarlama

- Müşterilerin satın alma örüntülerinin belirlenmesi,
- Müşterilerin demografik özellikleri arasındaki bağlantıların bulunması,
- Posta kampanyalarında cevap verme oranının artırılması,
- Mevcut müşterilerin elde tutulması, yeni müşterilerin kazanılması,
- Pazar sepeti analizi,
- Müşteri ilişkileri yönetimi,
- Müşteri değerlendirme,
- Satış tahmini.

Bankacılık

- Farklı finansal göstergeler arasında gizli korelasyonların bulunması,
- Kredi kartı dolandırıcılıklarının tespiti,
- Kredi kartı harcamalarına göre müşteri gruplarının belirlenmesi,
- Kredi taleplerinin değerlendirilmesi,
- Sahte davranışları tanımlama,

- Sadık müşterilerin tespiti.

Finans

- Sigortalamak için uzman sistem kuralları keşfetme,
- Portföy yönetimi,
- Para değeri değişimini tahmin etme.

Sigortacılık

- Yeni poliçe talep edecek müşterilerin tahmin edilmesi,
- Sigorta dolandırıcılıklarının tespiti,
- Riskli müşteri örüntülerinin belirlenmesi.

Mühendislik

- Simülasyon ve analiz,
- Örüntü tanıma,
- Sinyal işleme.

Internet

- Arama motorları,
- Web üzerinden pazarlama,
- Web madenciliği.

Tıp

- Hasta davranışını karakterize etme,
- Farklı hastalıklar için başarılı tedavileri tanımlama,
- Hipotez keşfi, tahminleme, sınıflandırma, tanı koyma.

VM'nin işletmelerde kullanım amaçlarına örnekler:

- Bir işletme kendi müşterisi iken rakibine giden müşterilerle ilgili analizler yaparak rakiplerini tercih eden müşterilerinin özelliklerini elde edebilir ve bundan yola çıkarak gelecek dönemlerde kaybetme olasılığı olan müşterilerin kimler olabileceğini bulunarak; onları kaybetmemek, kaybettiklerini geri kazanmak için strateji geliştirebilir.

- Ürün ve hizmette hangi özelliklerin ne derecede müşteri memnuniyetini etkilediği, hangi özelliklerinden dolayı müşterinin bunları tercih ettiği ortaya çıkarılabilir.
- Müşterilerin kredi riskleri hesaplanarak hangi müşterilerin kredi riskinin yüksek olduğu, hangi müşterilerin geri ödemesini zamanında yapamayabileceği kestirilebilir. Kredi kartı ödemelerini aksatan, gecikmeli olarak yapan veya hiç yapmayanların özelliklerinden yola çıkılarak bundan sonra aynı duruma düşebilecek muhtemel kişiler saptanabilir.
- En karlı mevcut müşteriler saptanarak, potansiyel müşteriler arasından en karlı olabilecekler belirlenebilir. Karlı müşteriler tespit edilerek onlara özel kampanyalar uygulanabilir. En masraflı müşteriler daha masrafsız müşteri haline dönüştürülebilir. Örneğin en çok bankacılık işlemi yapanlar ortaya çıkarılıp, bunlar şube bankacılığı yerine daha masrafsız Internet bankacılığına yönlendirilebilir.
- Bir ürün ve hizmetle ilgili bir kampanya programı oluşturmak için hedef kitlenin seçiminden başlayarak bunun hedef kitleye hangi kanallardan sunulacağı kararına kadar olan süreçte VM kullanabilir (Özmen, 2002).

2.3.3. VM uygulamalarında karşılaşılan problemler

Küçük veri kümelerinde hızlı ve doğru bir biçimde çalışan bir sistem, çok büyük veritabanlarına uygulandığında tamamen farklı davranabilir. Bir VM sistemi tutarlı veri üzerinde mükemmel çalışırken, aynı veriye gürültü eklendiğinde kayda değer bir biçimde kötüleşebilir. İzleyen kesimde günümüz VM sistemlerinin karşı karşıya olduğu problemlere değinilmiştir.

Örneklemin büyük olması, örüntülerin gerçekten var olduğunu göstermesi açısından bir avantajdır; ancak böyle bir örneklemden elde edilebilecek olası örüntü sayısı çok büyüktür. Bu yüzden, VM sistemlerinin karşı karşıya olduğu en önemli sorunlardan biri, veritabanı boyutunun çok büyük olmasıdır. Bir çözüm olarak veri kümesinin örneklemi alınarak işlemler yürütülebilir.

Bir veritabanında gürültü veri girişi veya toplanması sırasında oluşan sistem dışı hatalardır. Bu durum, bir VM yönteminin kullanılan veri kümesinde bulunan gürültülü

verilere karşı daha az duyarlı olmasını gerektirir. Gürültülü verinin yol açtığı problemler tümevarımsal karar ağaçlarında uygulanan metotlar bağlamında kapsamlı bir biçimde araştırılmıştır. Veri kümesi gürültülü ise sistem tutarsız veriyi tanımalı ve ihmal edilmelidir (Chan ve Wong, 1991; Quinlan,1986).

Kullanılan tabloda yer alan tüm kayıtlar aynı sayıda niteliğe, niteliğin değeri boş olsa bile, sahip olmalıdır. Veritabanlarında birincil anahtarda yer almayan herhangi bir niteliğin değeri "boş değer" olabilir. İlişkisel veritabanlarında boş değer; bilinmeyen, uygulanamaz ve bilinmeyen veya uygulanamaz olacak biçimde üçe ayıran bir yaklaşım bulunmaktadır. Boş değerli nitelikler, bir veri kümesinde bulunuyorsa, ya bu değerler tamamıyla ihmal edilmeli ya da bu niteliğe olası en yakın değer atanmalıdır (Quinlan, 1986).

Veriler; kurum ihtiyaçları göz önünde bulundurularak düzenlenip, toplandığından, mevcut verideki eksiklikler nedeniyle gerçek hayatı yeterince yansıtmayabilir. Örneğin, hastalığın tanısını koymak için kurallar sadece çok yaşlı insanların belirtilerinin bulunduğu bir veri kümesi kullanılarak üretildiğinde; bu kurallara dayanılarak bir çocuğa tanı koymak pek doğru olmaz. Bu gibi koşullarda bilgi keşif modeli, ancak belirli bir güvenilirlik derecesinde tahmini kararlar alabilir.

Veri kümesi, eldeki probleme uygun olmayan artık nitelikler içerebilir. Bu nedenle hedef problemi tanımlamak için yeterli ve gerekli olan niteliklerin seçilmesi işlemi arama uzayı küçültülür. Bu işlem, sınıflandırma işleminin kalitesini de artırır (Almuallim vd., 1991; Kira vd., 1992).

Kurumsal çevrim-içi veritabanları dinamiktir, yani içeriği sürekli olarak değişir. Bu durum, bilgi keşif teknikleri için önemli sakıncalar doğurmaktadır. İlk olarak sadece okuma yapan ve uzun süre çalışan bir bilgi keşif tekniği, bir veritabanı uygulaması olarak mevcut veritabanı ile birlikte çalıştırıldığında mevcut uygulamanın performansını ciddi ölçüde düşürür. Diğer bir sakınca ise, veritabanında bulunan verilerin kalıcı olduğu varsayılarak, çevrim-dışı veri üzerinde bilgi keşif metodu çalıştırıldığında, değişen verinin elde edilen örüntülere yansıması gecikecektir (Vahaplar ve İnceoğlu, 2002).

2.3.4. Uygulamada kullanılan VM araçları

Ticari VM araçları için geniş bir saha vardır. Bir ucunda oldukça karmaşık ve pahalı şirket seviyesinde birçok görevleri gerçekleştirmek için çok sayıda tekniği birleştiren paketler; diğer ucunda ise basit ve ucuz sadece bir tekniği sınırlı kapasitede kullanabilen masaüstü ürünleri olmak üzere birçok seçenek bulunmaktadır.

Ticari VM araçları temelde üç bölümde incelenebilmektedir (Brachman vd., 1996; Ge, 1998; *Data Mining* '99: *Technology Report*, 1999):

- **Üst seviye (şirketler için) VM araçları:** Bunlar, karmaşık ve geniş ölçekli problemleri doğru analiz etmekle ilgilenen yüksek derecede uyarlanabilir ve karmaşık araçlardır. Büyük kuruluşlar, yüksek fiyat ve düşük kar payı ile büyük rekabet yaşanan pazarlama yönetiminde önemli miktarlarda para harcamaktadırlar. Bu miktar genel giderler dışında 100.000 US$'ı geçmektedir. Bunlara HyperParallel, Thinking Machine, NeoVista ve IBM in ürünleri örnek olarak verilebilir.
- **Orta seviye VM araçları:** Bunlar, orta-büyük arası büyüklükte önemli miktarda madencilikte kullanılabilecek veriye sahip işletmelerin karmaşık analiz problemlerini çözmek için kullanılırlar. 10.000–100.000 US$ arası fiyatla satın alınabilen ürünlerdir. DataMind ve SAS Ins. bu gruba hitap eden araçlar üretir.
- **Alt seviye (masaüstü) VM araçları:** Belirli bir alana özgü sınırlı kapasitede ürünlerdir. Bunların kullanılmasında genellikle çok az teknik birikim yeterlidir. Sınırlı karmaşıklıkta veriler incelenebilir ve çok kuvvetli tahminleyici modeller oluşturulamamaktadır. 1.000–10.000 US$ arası fiyata sahiptirler. Business Objects, Cognos ve SPSS ürünleri bu gruptadır.

VM araçlarını birbirlerinden ayıran temel özellik ve ölçütler, şu şekilde sıralanabilir:

- *Karmaşıklık*[17], geliştirilen VM aracının kullanıcı açısından kullanım zorluğunu ölçer. VM uygulamalarının gerçekleştirildiği araç için kullanım kolaylığı sağlaması ve yoğun bir eğitime ihtiyaç duyurmaması beklenir.

---

[17] Complexity

- *Esneklik*[18], VM aracında kaç farklı tekniğin gömülü olduğunu belirler. VM aracının farklı görevleri gerçekleştirmesi, denetleme mekanizmasında daha iyi olabilmektedir. Fakat bazı özel durumlarda belirli bir göreve atanmış VM araçları tercih sebebi olabilir. Bir VM aracı, problemleri çözmek için gerekli teknikleri kapsayan VM süreci ile beraber diğer VBK adımlarını içermesi ideal olanıdır.
- *Güçlülük*[19], bir VM aracı için sistem mimarisini tanımlar. Bir VM aracı güçlü ise, en ileri teknoloji ve gelişmiş işlemcileri kullanan en son mimaride çalışabilme kabiliyetindedir.
- *Ölçeklenirlik*[20], sistem genişlerken performansının da aynı paralellikte artırılabilmesi anlamındadır.

Goebel ve Gruenweld, (1999), ticari olarak pazarlanan 43 farklı VM ve VBK araçlarını *genel ürün karakterleri*, *veritabanı bağlantısı* ve *VM karakteristikleri* adı altında üç farklı sınıfta ve her bir sınıfta da çok sayıda özellik değerleri ile bu araçların kabiliyetlerini göstermektedir.

2.3.5. VM işlevleri ve kullandığı teknikler

VM süreci sonunda elde edilen örüntüler, kurallar biçiminde ifade edilir. Elde edilen kurallar ya iki değişkenin birliktelik derecesini gösterir ya veriyi önceden tanımlanmış sınıflara paylaştırır ya da veriyi tanımlayan sonlu sayıda kümeye ayırır. Bu kurallar veri üzerinde belirli bir tekniğin (algoritmanın) yinelenmesiyle elde edilir. Elde edilen bilginin kalitesi, veri analizi için kullanılan algoritmaya büyük ölçüde bağlıdır.

VM algoritmaları, genel olarak iki grupta toplanır. Bunlar doğrulamaya dayalı[21] ve keşfe dayalı algoritmalardır[22].

Doğrulamaya dayalı VM algoritmasında kullanıcı tarafından bir hipotez öne sürülür ve seçilen veri kümesinde hipotez doğruluğu test edilir. Sonuç pozitif ise işlem

---

[18] flexibility
[19] Powerfulness
[20] Scalability
[21] Verification driven
[22] Discovery driven

sonlandırılır. Aksi taktirde yeni bir sorgu oluşturulur. İşlem; sonuçlar, hipotezi doğrulayana kadar veya kullanıcının kullanılan veriler hakkında geçerli bir veri olmadığına karar verinceye kadar devam eder. Hipotez, ya mantıksal bir kural ya da mantıksal bir ifade ile gösterilir. Her iki biçimde de seçilen veritabanındaki nitelik alanları kullanılır. X ve Y birer mantıksal ifade olmak üzere "EĞER X İSE Y" biçiminde bir hipotez öne sürülebilir. Öne sürülen hipotez genellikle belirli bir örüntünün veritabanındaki varlığıyla ilgili bir tahmindir. Bu şekilde çok az yeni bilgi elde edilebilir. Bu tür analizleri gerçekleştirmek için SQL ve OLAP kullanılır. Doğrulamaya dayalı VM algoritmalarının en yaygın olarak kullanıldığı yerler, istatistiksel ve çok boyutlu analizlerdir.

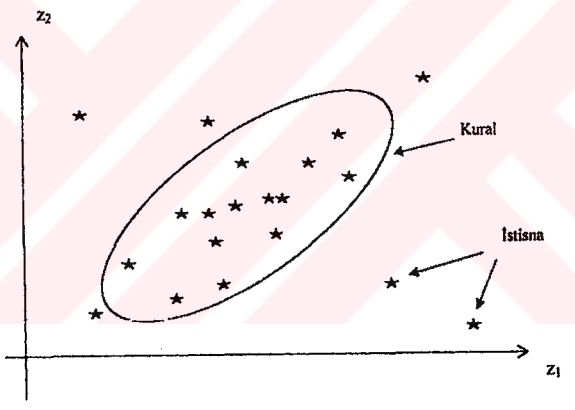
Keşfe dayalı VM algoritmalarında sistem, veri üzerinde sıklıkla meydana gelen örüntüleri arar, eğilimleri ortaya çıkarır ve veri hakkında genellemeler üretir. Keşif, kullanıcının çok az rehberliği ile tamamlanır. Burada tahminleyici modeller kullanılır ve sonuçları bilinen verilerden hareket edilerek bir model geliştirilmesi ve kurulan bu modelden yararlanılarak sonuçları bilinmeyen veri kümeleri için sonuç değerlerin tahmin edilmesi amaçlanır. Örneğin bir banka önceki dönemlerde vermiş olduğu kredilere ilişkin gerekli tüm verilere sahip olabilir. Bu verilerde bağımsız değişkenler; kredi çeken müşterinin özellikleri, bağımlı değişken değeri ise kredinin geri ödenip ödenmediğidir. Bu verileri uygun olarak kurulan model, daha sonraki kredi taleplerinde müşteri özelliklerine göre verilecek olan kredinin geri ödenip ödenmeyeceğinin tahmininde kullanılmaktadır.

Keşfe dayalı VM'de tanımlayıcı[23] ve tahminleyici[24] VM olmak üzere iki gruba ayrılır. Tanımlayıcı VM, veri kümesini tanımlar ve verinin ilginç genel özelliklerini sunar. Tahminleyici VM ise bir veya daha fazla model kümeleri oluşturur, mevcut veri kümesinde çıkarsama gerçekleştirir ve yeni veri kümelerinin davranışını tahmin etmeye çalışır (Agrawal vd., 1996, Chen vd., 1996).

---

[23] Descriptive
[24] Predictive

Tanımlayıcı modellemeler, önceden fikir yürütülmemiş veri içerisinde neler olduğunu daha iyi anlamamızı sağlayan bilgiler sunarlar. Program, büyük veritabanlarında ilgi çekici örüntüleri bulurken inisiyatif alır; çünkü kullanıcının ancak doğru soruları sorarak bulabileceği ve pratik olarak düşünemeyeceği kadar çok sayıda örüntü vardır. Keşfin gücü ve kullanışlılığı, keşfedilen bilginin zenginlik ve kalitesi ile ilgilidir. Tanımlayıcı VM'de hiç bir değişken hedef olarak seçilmeyerek tüm değişkenler arasında bazı ilişkiler kurmak amaçlanır. X-Y aralığında geliri ve iki veya daha fazla arabası olan çocuklu aileler ile, çocuğu olmayan ve geliri X-Y aralığından düşük olan ailelerin satın alma örüntülerinin birbirlerine benzerlik gösterdiğinin belirlenmesi tanımlayıcı modellere bir örnektir. Veri özetleme ve görselleştirme, bağıntı analizi ve kümeleme tanımlayıcı modellemeye girer.



Şekil 2.6 Veri görselleştirme grafiği (Alpaydın, 2000)

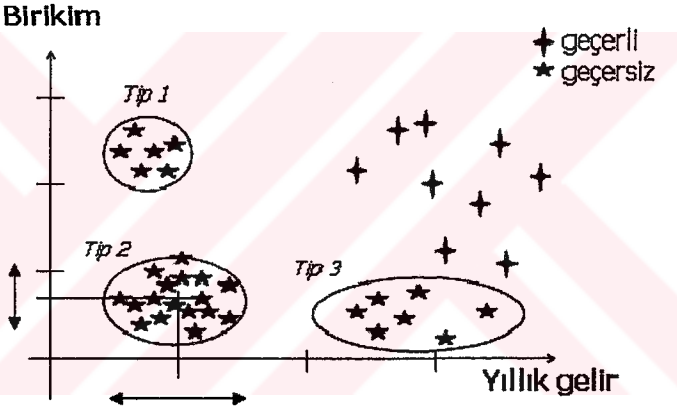
Veri özetleme ve görselleştirme,[25] veriyi daha iyi anlamamıza yardım eden tekniklerdir. *Özetleme*, tanımlayıcı istatistikler kullanılarak verinin basitçe tanımlanmasıdır. *Görselleştirme*; histogramlar, çubuk ve noktasal grafikleri kullanarak verinin basitçe grafiksel gösterimidir. Genel olarak sınır dışı durumlar, genel eğilim ve

[25] Summarization and visualization

ilişkilere bakmak ve projenin başlangıcında çeşitli kararlar vermede yardım etmesi için veri hakkında öz bilgi edinmede kullanılırlar.

Bağıntı analizi,[26] kayıtlar ile geliştirilen modeller arasındaki ilişkileri, ilişkilerdeki örüntülere dayalı olarak takip eder. Grafik teorisinden adapte edildiğinden güçlü görselleştirmeye sahiptir. Bu da işlemin anlaşılabilirliğini kolaylaştırır. Bununla beraber bir çok tür probleme uygulanamamaktadır.

Kümeleme algoritması[27], veritabanını alt kümelere ayırır. Her bir kümede yer alan elemanlar, dahil oldukları grubu diğer gruplardan ayıran ortak özelliklere sahiptir (Michalski vd., 1983). Sadece kümeleri bulmak bunların taşıdıkları anlamı anlamak için yeterli olmayabilir. Bu gibi durumlarda ek olarak başka yöntemler de uygulamak gereklidir.



Şekil 2.7 Kümeleme

Kümeleme modellerinde amaç, Şekil 2.7' de görüldüğü gibi küme üyelerinin birbirlerine çok benzediği, ancak özellikleri birbirlerinden çok farklı olan kümelerin bulunması ve veritabanındaki kayıtların bu farklı kümelere bölünmesidir. (Paul vd., 2004). Yaygın kullanım alanları nüfusbilimi, astronomi vb. dir.

---

[26] Link Analysis
[27] Clustering

Tahminleyici modelleme kullanıcıya kayıtları bazı bilinmeyen alan değerleri ile birlikte sunmaya izin verir ve sistem, veritabanından önceden keşfedilen örüntülere dayalı olarak bilinmeyen değerleri tahmin eder. Denetimli VM'de amaç; hazır veriyi kullanarak ilgilenilen belirli bir değişkeni, kalan diğer kullanışlı veri cinsinden tanımlayan model oluşturmaktır. Sınıflandırma, kestirim ve tahminleme tahminleyici modellemeye girer.

Sınıflandırma,[28] bir sınıfı diğerinden ayıran örüntüleri keşfeder. Sınıflandırma, yeni bir veri elemanını daha önceden belirlenmiş sınıflara atamayı amaçlar (Weiss vd., 1991). Veritabanında yer alan kayıtlar, bir sınıflandırma fonksiyonu yardımıyla kullanıcı tarafından belirlenmiş ya da hedef niteliğinin bazı değerlerine göre anlamlı alt sınıflara ayrılır. Yaygın kullanım alanları, banka kredisi onaylama işlemi, kredi kartı sahteciliği tespiti ve sigorta risk analizidir. Sınıflandırma algoritmaları iki şekilde kullanılır:

- *Hedef Değişken ile Sınıflandırma:* Veritabanındaki kayıtlar, belirlenen bir hedef değişkeninin değerlerine göre sınıflara ayrılır. Aynı sınıfta yer alan kayıtlar, karar değişkeninin değeri açısından özdeştir.
- *Örnek ile Sınıflandırma:* Bu türdeki sınıflandırmada veritabanındaki değerler iki kümeye ayrılır. Kümelerden biri pozitif, diğeri ise negatif değerler içerir.

Kestirim,[29] sadece kesikli sonuçlar ile ilgilen sınıflandırma algoritmasının aksine, sürekli değerler içeren sonuçlar ile ilgilenir. Kestirim, verilen giriş değerleri ile bazı bilinmeyen sürekli değişkenler için değerler ortaya çıkarır.

Zaman serisi öngörüsü[30] olarak da adlandırılan tahminleme[31] sınıflandırma veya kestirim tekniklerine çok benzemektedir. Ayrıldığı nokta, kayıtların sınıflandırılması bazı tahmin edilen gelecek davranışlarına veya kestirilen gelecek değerlerine göre olmasıdır. Buradaki önemli nokta, bu değerlerin zamanla olan bağımlılıklarıdır.

---

[28] Classification
[29] Estimation
[30] Time-series forecasting
[31] Prediction

VM'nin yukarıda belirtilen iki ana türünde de kullanılan teknikler tamamen yeni değildir. Bunlar YZ, grafik teorisi[32], MÖ, olasılık ve istatistik gibi diğer disiplinlerden adapte edilmişlerdir.

Bunlardan başka eşleştirme kuralları[33], karar ağaçları[34], genetik algoritmalar[35], en yakın komşuluk ilişkisi[36], sinir ağları[37] ve kural çıkarma[38] gibi VM'de sıkça kullanılan teknikler; kullanıcıya kısa zaman içerisinde çok sayıda veriyi inceleme gücü, veri ve sonuçları açıkça sunma kabiliyeti verirler.

Eşleştirme kuralları, aşağıda sunulan örneklerde görüldüğü gibi, eş zamanlı olarak gerçekleşen ilişkilerin tanımlanmasında, bağıntı analizi içerisinde kullanılır (Agrawal vd., 1993).

- *Müşteriler kola satın aldığında, 0.75 ihtimalle patates cipsi de alırlar,*
- *Az yağlı peynir ve yağsız yoğurt alan müşteriler. 0.85 ihtimalle diyet süt de satın alırlar.*

Yaygın kullanım alanları; katalog tasarımı, mağaza ürün yerleşim planı, müşteri kesimleme, telekomünikasyon vb.dir.

Ardışık örüntü keşfinde ise eş zamanlı olarak değil, belirli bir zaman aralığında sıklıkla gerçekleşen olaylar kümelerini bulmak amaçlanır (Agrawal vd.,1993). Ardışık örüntü örnekleri aşağıdakiler gibi olabilir:

- *Bir yıl içinde Orhan Pamuk'un "Benim Adım Kırmızı" romanını satın alan insanların %70'i Buket Uzuner' in "Güneş Yiyen Çingene" adlı kitabını satın almıştır.*
- *X ameliyatı yapıldığında, 15 gün içinde 0.45 ihtimalle Y enfeksiyonu oluşacaktır.*

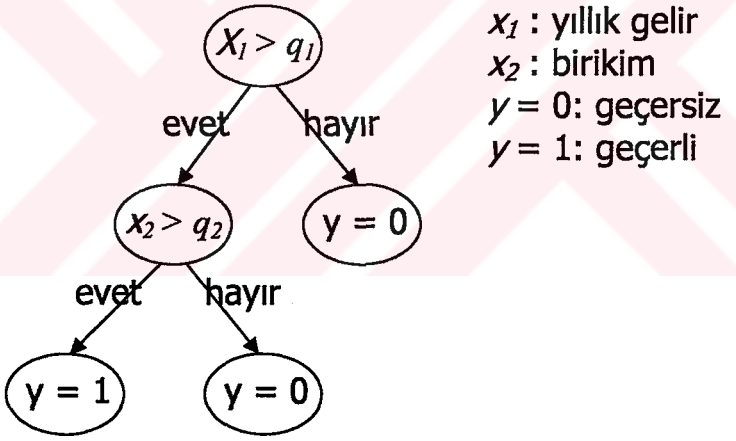
Satın alma eğilimlerinin tanımlanmasını sağlayan eşleştirme kuralları ve ardışık zamanlı örüntüler; pazarlama amaçlı olarak sepet analizi[39] adı altında VM'de yaygın olarak kullanılmaktadır. Bununla birlikte bu teknikler; tıp, finans ve farklı olayların

---

[32] Graph theory
[33] Association Rules
[34] Decision Trees
[35] Genetic Algorithms
[36] K- Nearest Neighbour (K-NN)
[37] Neural Network
[38] Rule Induction
[39] Basket Analysis

birbirleri ile ilişkili olduğunun belirlenmesi sonucunda değerli bilgi kazanımının söz konusu olduğu ortamlarda da önem taşımaktadır.

Karar ağaçları, tahminleyici modellemede çok iyi çalışan modeller oluşturabilen algoritmalardır. Veri kümesindeki her bir sütunun, hedef sütunun sonucunu nasıl etkilediğine bakarak en kuvvetli ilişkili sütunları bulur ve bu sütunları kullanarak ayrım serileri oluşturur. Bu ayrımların her birine *düğüm* adı verilir ve ağaç yapısında görüntülenebilir. En üst düğüm (*kök*), tüm popülasyon üstünde tahmin edilmiş niteliğin dökümünü tanımlar (Paul vd., 2004). Şekil 2.8'de ilk ayrım yıllık gelirin q_1 değerinden büyük olup olmaması; ikinci ayrım, birikimin q_2 değerinden büyük olup olmamasıdır. Kişi, yıllık geliri q_1 den büyük ve var olan birikimi q_2 den büyük ise geçerli kritere sahip; aksi halde geçerli kriterleri sağlamamış olarak düşünülür. Karar ağaçları, genellikle kümeleme, sınıflandırma ve tahminleme gibi farklı VM teknikleri içerisinde kullanılır.



Şekil 2.8 Karar Ağacı

Karar ağaçları, tez uygulamasının model oluşturma ve gösteriminde de kullanıldığından 3.bölümde detaylı olarak bilgi verilmiştir.

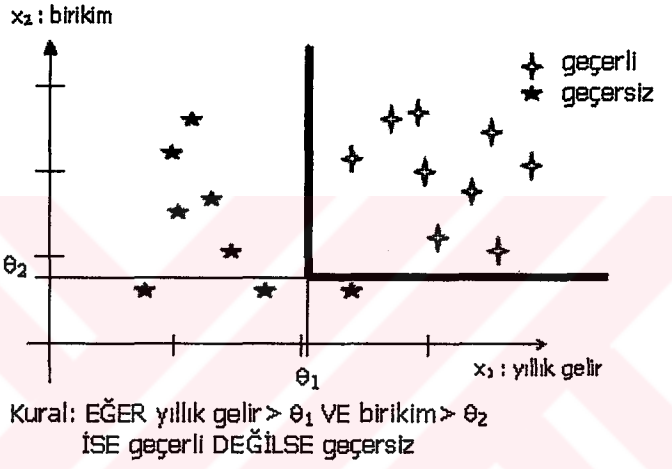
Genetik algoritmalar, en çok bilinen ve en yaygın olarak kullanılan evrimsel algoritmalardır. Ağır hesaplama problemlerinde, özellikle de sınıflandırmada tercih

edilir. Evrim teorisinde yer alan "Bir neslin üyeleri kendi karakterlerini en iyi duruma getirene kadar gelecek nesle aktarmak için yarış içerisinde olur." düşüncesinden esinlenilerek aynı düşüncenin bilgisayar ile donatılmış bir versiyonu olarak "genetik" algoritmalar geliştirilmiştir. Aktarılan bilgi, kromozomlar içerir. Kromozomlar, modeli oluşturmak için parametreler içerir. Zamanla, ufak program parçaları olan organizmalar; bilgisayarda optimize edilir, belirli bir problemi çözümleme performanslarında gelişme sağlanır ve yüksek derece yetenek kazandırılır. Sonra topluluktaki (simule edilmiş evrim geçiren organizma topluluğu) bireyler *evrimleştirilir*. Sonunda, belirli bir problemi çözümlemede en iyi durumda bulunan simule edilmiş organizmaları tutup ve daha az başarılı olanları bilgisayar hafızasından silen *seçim* işlemi yer alır. Genetik algoritmalar, genellikle verilerdeki örüntüleri bulmak yerine, diğer VM algoritmalarını optimize etmeye yardımcı olmak için kullanılırlar. Genetik algoritmalar, açıklanabilir sonuçlar üretirler. Bu nedenle sonuçlar kolay bir şekilde uygulanabilmektedir (Berson vd., 1997; Ge, 1998; De Jong, 1999,).

Yeni bir problemi çözerken, daha önceden çözülmüş problemlerin çözümlerine bakmak çoğu kez daha iyidir. En yakın komşuluk (K-NN), buna benzer bir yöntemi kullanan bir sınıflandırma tekniğidir. Bu sınıflama, bir veri kümesi içinde k sayıda kaydın, her bir kayıtla karşılaştırılmasına dayanır. Yeni bir durumun hangi sınıfın içinde yer alacağına k sayıda en çok benzer durumları veya komşuları inceleyerek karar verir. Her sınıf için durum sayısını sayar ve yeni durumu benzer yani komşularının çoğunun ait olduğu sınıfa atar (Data Mining '99: Technology Report, 1999).

VM'de en eski ve en sıkça kullanılan tekniklerden biri sinir ağlarıdır (http://www.datamining.com/dm-technology.htm, 1997; Berry vd., 2000). Yapay sinir ağları olarak da isimlendirilir. Berson ve Smith (1997), yapay sinir ağlarını çok sayıda tahmin edici değişkene sahip karmaşık problemler için geniş veritabanlarında tahminleyici modeller oluşturmak için karmaşık örüntü algılama ve makine öğrenimi algoritmalarını uygulayan bilgisayar programları olarak tanımlar. İnsanın yaptığına benzer şekilde örüntü algılamayı öğrenir ve daha iyi tahminler yapar. Fu (1999), bunların biyolojik sinir ağlarına nasıl benzediğini sistem tanıtımı ile birlikte geniş bir şekilde anlatmaktadır.

Kural çıkarma, farklı durumları sınıflandırmak için "If Then" kurallar kümesi türeten bir tekniktir. Karar ağaçları da kurallar kümesi çıkarır; ancak kural çıkarma tekniklerinde üretilen kurallar kümesinde; karar ağaçlarından farklı olarak, bir ağaç şeklinde gösterilmeye ihtiyaç duyulmaz, tüm olası durumları kapsamaz ve bu kurallar bazen kendi tahminleri ile ters düşebilirler. Bu durumda takip edilecek kuralı seçmek gerekir. Bu tür karmaşaları çözmekte kullanılan genel bir yöntem kurallara güven ataması yapmak ve daha güvenilir olanı kullanmaktır (Berson vd., 1997; Ge, 1998).



Şekil 2.9 Kural Çıkarına Grafiği (Alpaydın, 2000)

2.3.6. VM'yi etkileyen eğilimler

Vahaplar ve İnceoğlu (2002)'na göre VM' yi temelde beş ana harici eğilim etkiler :

- **Veri:** VM'nin bu kadar gelişmesindeki en önemli etkendir. Son yirmi yılda sayısal verinin hızla artması, VM'deki gelişmeleri hızlandırmıştır. Bu kadar fazla veriye bilgisayar ağları üzerinden erişilebilmektedir. Diğer yanda bu verilerle uğraşan bilim adamları, mühendisler ve istatistikçilerin sayısı fazla artmamaktadır. Bu nedenle, verileri analiz etmede yeni yöntem ve teknikler, geliştirilmektedir.
- **Donanım:** VM, sayısal ve istatistiksel olarak büyük veri kümeleri üzerinde yoğun işlemler yapmayı gerektirir. Gelişen bellek ve işlemci hızı kapasitesi sayesinde,

birkaç yıl önceye kadar madencilik yapılamayan veriler üzerinde çalışmak şu anda mümkün hale gelmiştir.

- **Bilgisayar ağları:** İnternet, yakın bir gelecekte yaklaşık 155 Mbits/sn' lik hatta belki de daha da üzerinde veri akış hızı sağlayabilecektir. Bu da günümüzde kullanılan bilgisayar ağlarındaki hızın 100 katından daha fazla bir sürat ve taşıma kapasitesi demektir. Buna bağlı olarak, VM'ye uygun ağların tasarımı da yapılmaktadır. Böyle bir bilgisayar ağı ortamı oluştuktan sonra, dağıtık verileri analiz etmek ve farklı algoritmaları kullanmak mümkün olacaktır.
- **Bilimsel hesaplamalar:** Günümüz bilim adamları ve mühendisleri, simülasyonu teori ve deneyden sonra bilimin üçüncü yolu olarak görmektedirler. VM ve bilgi keşfi, bu üç metodu birbirine bağlamada önemli rol almaktadır.
- **Ticari eğilimler:** Günümüzün işletme yönetiminde sunulan servis ve hizmet kalitesi ve iş akış hızını artırmak hedeflenmektedir. Bütün bunları yaparken de minimum maliyet ve en az insan kaynağı harcaması göz önünde bulundurulmalıdır. Bu tür hedef ve kısıtların yer aldığı iş dünyasında VM, temel teknolojilerden biri haline gelmiştir. Çünkü VM sayesinde müşterilerin ve müşteri faaliyetlerinin meydana getirdiği fırsatlar daha kolay tespit edilebilmekte ve riskler daha açık görülebilmektedir. Dolayısı ile maliyetin azalmasına karşılık, hitap edilen müşteri sayısındaki artış, yüksek karı getirecektir.

2.3.7. VM'de geleceğe yönelik yaklaşımlar

Rigdon ve Bacon (1998), ideal VM araçlarında bulunması gereken özellikleri açıklamaktadır. Buna göre; VM araçları, insanın analiz etme ile makinenin hesaplama gücünü birleştirir. Tamamen otomatikleşmiş olmakla beraber, hangi bulguların iş platformunda ilgi çekici olduğunu tanımlamada yardımcı olan analizör ve yöneticileri devre dışı bırakmaz. Kısa zaman aralığında harekete geçirecek çözümler sunarlar. Dolayısı ile ölçeklenebilir yapıda, diğer ilişkili veritabanı ürünleri ile entegre durumda ve kullanıcı dostudur.

Mitchell (1999)'e göre; günümüzde VM araçları için önemli sınırlamalar bulunmaktadır. İlk olarak bunlar, verilerin sadece sayısal ve sembolik karakterler

taşıdığı; metin, görüntü özellikleri ve ham sensör verileri içermediğini farz eder. İkincisi, veriler tek bir veritabanına belirli bir VM görevi ile dikkatlice toplandığı farz edilir. Üçüncü olarak, tamamen otomatikleşmeye meyillidir ve bu nedenle veri düzenliliklerinin anahtar durumlarında bilgili birinin rehberliğine izin vermesine izin vermekte başarısız kalmaktadır ve son olarak VA ve görselleştirme araçları bütünleşmesi için gerek duyulan çabayı azaltmak için gelişmiş özelliklere ihtiyaç duyar.

Mitchell (1999), gösterilen bu sınırlamalara rağmen, VM'ye karşı güçlü bir ticari ilginin bulunmasında eğilimleri kuvvetlendiren üç unsura işaret etmektedir. Bunlar:

- Daha doğru öğrenen, çeşitli İnternet ve İntranet üzerinde yer alan daha çok uygun veri kaynaklarındaki veriyi kullanabilen ve çalışma esnasında daha fazla insan girişine izin veren yeni makine öğrenimi algoritmalarının gelişmesi,
- Bu algoritma bilgilerinin veritabanı yönetim sistemleri[40] ile bütünleşebilmesi,
- VM teknolojisi ve buna yardım eden veri toplama ve ambarlama, tarihsel veriyi delile dayalı karar verme sistemlerinin kullanımında gözle görülen artış.

Yukarıda belirtilen düşünceler ve eğilimler ışığında gelecekte geliştirilecek VM araçlarında bulunacağı tahmin edilen kabiliyetleri ve getireceği etkileri Fayyad vd. (1996), Mitchell (1999) ve Munakata (1999) özetlemektedir. Buna göre; çeşitli kaynaklardan büyük, daha geniş kapsamlı, aşırı basitleştirilmemiş, boş, gürültülü ve sayısal ses ve resim gibi gerçek dünya türünde veriler ile ilgilenilecektir. Modelin eğitiminde çok sayıda veritabanından ve hatta web (ağ) alanından öğrenme gerçekleştirilecektir. İnsanın inisiyatifini ve seçiciliğini iptal etmeden madencilik sürecinin verimliliğini ve keşfedilen bilginin kalitesini artıran insan-bilgisayar arası etkileşim artacaktır. OLAP ve VA araçlarını bütünleştirerek farklı teknikleri kullanan melez sistemler kullanılarak sistemlerdeki zayıf noktalar telafi edebileceklerdir. Olası sonuçları tahmin etmede makine öğrenimi algoritmaları; gizli değişkenler ve bağımlılıkları tanımlayabilmek için çok karmaşık istatistiksel stratejiler geliştirilecektir. Şu anda kullanılan VM sistemlerinin çoğu önceden belirlenmiş veri kümesini pasif olarak kabul etmektedir. İlave kullanışlı bilgiler elde edebilmek için aktif olarak en iyi

---

[40] Database management systems

deneyler gerçekleştirebilen yeni bilgisayar yöntemleri geliştirilecektir. VM teknolojisindeki ilerlemeler, sonunda yükseltilmiş veri analizi yetenekleri olarak sonuçlanacaktır. Böylece; karar vericilere veriyi daha etkin bir şekilde toplama, depolama ve analizlerini gerçekleştirebilme imkanı sunulacaktır. Bu nedenle; çoğu sektör, kendi verilerinden faydalanarak harekete geçirilebilir kararlar verebilmek için, VM uygulamalarını tercih edeceklerdir.

VM araçları işletmelere zamanında akıllı ve doğru kararlar vermelerini sağlarken, bu işletmelerin VM süreci boyunca başarıya ulaşabilmesi için bazı anahtar noktaları anlaması gerekmektedir. VM sürecinde başarı kazanmak için bazı önemli etkenler dikkate alınmalıdır. Hermiz (1999), bunları;

- Çözülmek istenen problemin açık bir şekilde tanımlanması,
- Takip edilen problemi destekleyen verilerin gerçek veri tipinde, yeterli kalitede ve yeterli miktarda olması,
- VM birçok parçalardan oluşan bir süreç olarak bunların birbirleriyle uygun bir şekilde bağlanması ve yönetiminin gerektiğinin farkında olunması,
- VM sürecinden öğrenmek için planlama yapılması,

olarak sıralamaktadır.

3. MATERYAL ve YÖNTEM

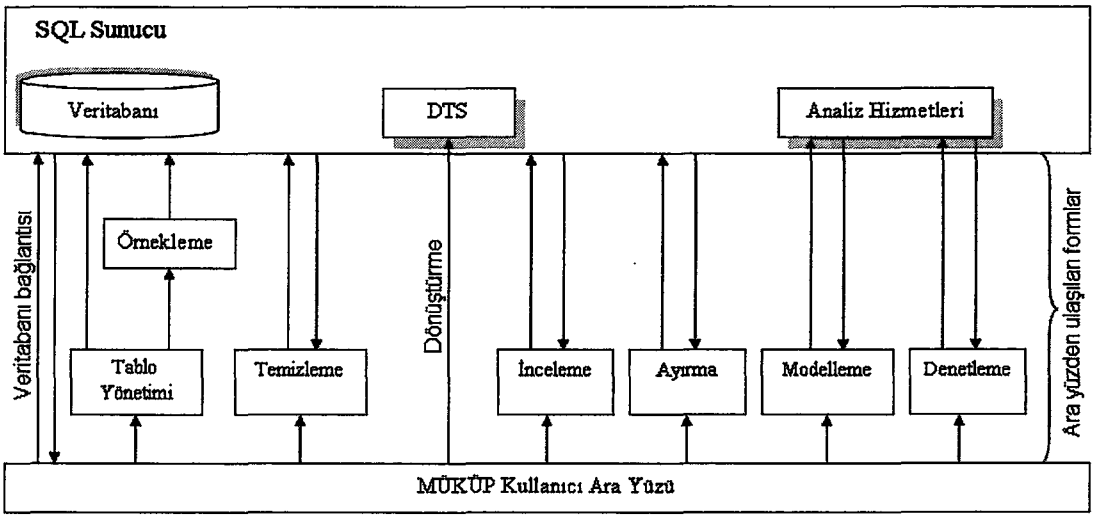
Bu tez çalışması, Muğla Üniversitenin bünyesindeki elektronik ortamda saklı tutulan öğrenci verileri üzerinde veri madenciliği içerisinde bir sınıflandırma tekniği olan karar ağaçları kullanılarak Muğla Üniversitesi öğrencilerinin lisans eğitimlerinde onları başarıya götüren kişisel özelliklerini bir profil olarak keşfetmek amacıyla gerçekleştirilmiştir.

Çalışma iki aşamadan oluşmaktadır. İlk olarak, öğrencilerin geçmiş kayıtları üzerinde çeşitli veritabanı sorguları gerçekleştirilmiştir. Bu sorgular neticesinde, öğrencileri daha iyi tanımamızı sağlayacak ve çalışmanın ilerleyen kesiminde katkı sağlayacak; öğrencilere ait çeşitli demografik bilgiler, üniversite öncesi eğitim bilgileri ve üniversitedeki eğitimleri süresince oluşturdukları bazı bilgiler elde edilmiştir. İkinci aşamada, keşfe yönelik bir çalışma olarak öğrenci sınıflandırması gerçekleştirilmiştir. Bu işlem sırasında öğrencilere ait dönem sonu ortalaması kategorize edilerek hedef değişken olarak belirlenmiştir. Öğrenciler, burada kişisel bilgileri ve dönem sonu ortalamalarına dayalı olarak başarılı veya başarısız olarak sınıflandırılmaktadır.

Çalışmada kullanılan Muğla Üniversitesi öğrencilerine ait veriler, üniversitenin Bilgi İşlem Dairesi ve Öğrenci İşleri Dairesi izni ile alınmıştır. Öğrencilere ait kişisel veya eğitim ile ilgili bilgiler içeren bu ham veriler, yerel bir veritabanında tutulmuştur. Çalışmada pilot olarak 1995 yılı ve sonrası İİBF öğrenci verileri kullanılmıştır. Veriler üzerindeki tüm bilgi keşfi süreci işlemleri, bu iş için geliştirilen MÜKÜP ile gerçekleştirilmiştir. MÜKÜP, SQL sunucu üzerinde; (SQL Server), Analiz Hizmetleri (Analysis Services) ve görselleştirme araçları ile ayrı ayrı gerçekleştirilebilecek görevleri bir arada tutmakta ve alt tabanda bu programların işlevlerini kullanmaktadır.

3.1. Muğla Üniversitesi Öğrenci Verileri Üzerinde Bilgi Keşfi Çalışmasının Tanıtımı

Bu bölüm, VM ve diğer VBK sürecine ait ilişkili görevlerin uygulanışı ve bu görevleri bir arada gerçekleştirmek amacıyla özel olarak geliştirilen programın oluşturulması ve işletilmesinde izlenen metodolojiyi kapsar.



Şekil 3.1 MÜKÜP blok diyagram

Çalışmanın çözüm geliştirme aşamasında her biri bir VBK adımı ile birleşmiş görevleri tamamlamak için SQL sunucu ve Analiz Hizmetleri arasında bölünmüş bir görev paylaşım mekanizması geliştirilmiştir. Her bir görev tek başına SQL sunucu veya Analiz Hizmetlerinde ayrı ayrı gerçekleştirilebilmesine rağmen bu görevlerin bütününü bir arada tutan bir ortam olarak MÜKÜP (Muğla Üniversitesi Öğrenci Bilgi Keşif Ünitesi Programı) geliştirilmiştir. Bu yaklaşımın avantajları:

- VBK süreci için örnek teşkil etmek,
- Tüm görevlere bir programdan erişilebilmek,
- SQL sunucu ve Analiz Hizmetlerinin işlevselliklerine programlama teknikleri ile erişebilmektir.

Ek1-Form1, MÜKÜP'ün kullanıcı ara yüzünü göstermektedir. Kullanıcı ara yüzünde gösterilen basamaklar, 2. bölümde tanıtılan ve Şekil 2.2'de gösterilen VBK süreci ile oldukça eşleşmektedir. MÜKÜP, üç ana bölümden oluşmaktadır:

- Veritabanı bağlantısı,
- Veri hazırlama,
- Model oluşturma,

Bu bölümlerde bulunan her bir düğme bir görevi temsil etmektedir ve bu görev ile ilgili bir form veya sihirbaza götürür. Bölümler ve bölümlerdeki görevleri yerine getirmede sırayı gözetmek önemli olduğundan, ara yüz bunu ifade edecek şekilde soldan sağa doğru akış görünümünde tasarlanmıştır.

Şekil 3.1'deki blok diyagram, MÜKÜP'ün çalışma prensibini göstermektedir. Kullanıcı ara yüzünden 8 adet farklı görevlere yönelik formlara doğrudan veya dolaylı olarak ulaşılabilmektedir.

Veritabanı bağlantısı ile üzerinde çalışılan veri kümesi veya tabloya bağlantı kurulur. Bu nedenle programın çalışabilmesi için ilk şarttır. Tablo yönetimi formunda ve bu formdan ulaşılabilen örnekleme formunda tablolar üzerinde çeşitli düzenlemeler ve yedeklemeler yapılabilmektedir.

Veri hazırlama bölümünde; veri temizleme, veri dönüştürme ve veri inceleme görevleri yer almaktadır. Veri dönüştürme, ayrı bir form olarak değil, SQL Sunucu içerisinde yer alan DTS (Data Transformation Services) Import/Export sihirbazı kullanılarak gerçekleştirilmektedir.

Model oluşturma bölümünde; çalışılan veri kümesini eğitim ve test veri kümelerine ayırma, modelleme kısmında eğitim veri kümesini kullanarak modeller oluşturma ve denetleme kısmında ise test veri kümesi kullanılarak modelin geçerliliğini denetleme görevleri gerçekleştirilir. Bu görevlerin nasıl gerçekleştirildiği ilerleyen kısımlarda daha detaylı olarak tanıtılacaktır.

SQL Sunucu, veritabanı yönetim işlemlerinde ve Analiz Hizmetleri ise model oluşturma ve oluşturulan modelin test edilmesinde esas olarak işlevleri yerine getiren unsurlardır. Kullanıcı ara yüzünden ulaşılan formlar, kullanıcıya VBK sürecini kolay bir şekilde takip edebilme, veritabanını yönetim faaliyetlerini düzenli ve kolay bir şekilde gerçekleştirebilme ve SQL Sunucu ve Analiz Hizmetleri ile kurulması gerekli bağlantı ve bilgi alış-verişlerini doğrudan gerçekleştirebilme avantajları sağlamaktadır. Bu şekilde kullanıcı, SQL Sunucu ve Analiz Hizmetlerinden tamamen soyutlanarak işlemlerini çok daha kolay ve hata doğurmayacak bir şekilde gerçekleştirmektedir.

Programda, öğrenci verileri düşünülerek geliştirildiğinden belirli kısıtlamalar bulunmaktadır. Farklı problemlere çözüm getirebilmesi için veri hazırlama bölümüne

bazı eklentiler yapılmalı, karar ağacı haricinde farklı modelleme teknikleri ve kaldıraç grafiği haricinde denetleme teknikleri de dahil edilmelidir.

3.1.1. Çalışmada Kullanılan teknolojiler

Bu çalışmada aşağıdaki teknolojiler kullanılmıştır:

- Windows Server 2003

İşletim sistemi olarak veritabanı üzerinde iyi bir yönetim sağlayabilmek için tercih edilmiştir.

- SQL Server 2000

Asıl olarak kaynak verisini depolamada kullanılırken; aynı zamanda temizleme, inceleme ve ayırma işlemleri sırasında üretilen tabloları oluşturmak, muhafaza etmek ve yönetmekte kullanılmıştır.

- Analysis Services

VM modelini oluşturma ve depolamada ve modelin performansını test etmek için tahminleri oluşturmakta kullanılmıştır.

- Visual Basic 6.0

Microsoft Visual Studio 6.0 içinde yer alır. MÜKÜP'ün kodunu oluşturmak, görmek ve derlemekte kullanılmıştır.

- MDAC 2.8

Veriye erişmede ihtiyaç duyulmuştur.

- Angoss OLE DB for Data Mining Consumer Controls

Model ve test sonuçlarını görüntülemek için tercih edilmiştir.

MÜKÜP, Visual Basic ve birlikte gelen ADO (Microsoft ActiveX Data Objects) ve DSO (Decision Support Objects) programlama teknolojileri kullanılarak geliştirilmiştir. Tablolardaki verilere bakabilmek için FlexGrid denetimi; veriyi araştırmak için chart denetimi; VM modellerini görmek ve karşılaştırmak için üçüncü parti modelleme denetimleri kullanılmıştır.

Çözüm geliştirme aşamasında, verileri yönetme mekanizmasının ihtiyaç duyduğu şekilde yeni tablolar oluşturuldu, bazı tablolar iptal edildi veya değiştirildi. SQL sunucu bu konuda; veriyi düzenleme, saklama ve yönetme işlemlerinde ihtiyaç duyduğumuz

fonksiyonellikleri sağlamıştır. VM süreci ile ilgili bazı görevleri yerine getirirken yine SQL sunucu araçlarından faydalanılmıştır. Örnek olarak; dönüştürme adımında DTS Import/Export sihirbazı, tablo verisini alıp, sütun değerlerini dönüştürmeyi sağlar. DTS aynı zamanda, VM modeli tabanlı tahmin oluşturur ve sonuçlara göre hareket gerçekleştirir.

SQL sunucu ile yönetilen tablolardan model oluşturma aşamasında Analiz Hizmetleri devreye girer. Analiz Hizmetleri, SQL sunucu ile oluşturulan ilişkisel veri kaynağına göre model kurmakta kullanılmıştır.

3.1.2. Kurulum (Setup)

Kurulum, bir çok adımdan oluşan bir süreçtir. Öncelikle kullanılacak bilgisayar gerekli araç ve teknolojiler ile donatılmalıdır. 2.4 GHz işlemci hızı, 512 MB RAM bellek ile göreceli olarak yeterli hıza erişilebilir. Eğer bu değerlerin altına düşülürse işlemler, önemli ölçüde zaman alabilir.

MÜKÜP, aslında bağımsız bir program olmasına karşın; aşağıda kurulum sırası ile listelenen programların kurulumu üzerinde işlev yürütebilmektedir.

- Microsoft Windows Server 2003 SP1 (Enterprise Edition)
- Microsoft SQL Server 2000 SP3 (Developer Edition)
- Microsoft SQL Server 2000 Analysis Services SP3 (Developer Edition)
- MDAC 2.8
- Microsoft Visual Studio 6.0
- Angoss OLE DB for Data Mining Consumer Controls

3.2. Problem Tanımı

VBK sürecinin ilk adımı problem tanımlamadır. Burada belirli bir probleme yönelik çözüm geliştirilirken, ulaşılmak istenen hedef ve buna karşılık elde bulunan imkanlar, yapılacak analiz türleri ve analiz sonucu yapılacak test türleri belirlenir.

3.2.1. İş problemini tanımlama

Bu çalışmada hedeflenen, Muğla Üniversitesi öğrenci verileri üzerinden öğrenci kitlesini daha iyi tanımlamak ve eğitimdeki başarılarına etki eden etkenleri bulabilmektir.

Üniversitedeki öğrenci verileri ile çalışmaya başlamadan önce hangi kitle üzerinde bu işlemler gerçekleştirilecek; yani nasıl bir örneklem oluşturulacak sorusuna cevap verilmeliydi. Üniversitenin geçmişten bugüne tüm öğrenci verileri ile uğraşmak; veriler üzerindeki eksiklik ve tutarsızlıkları gidermeyi zorlaştıracağı ve farklı yapıda bölümleri içeren birçok fakülteye özel durumları göz ardı etmek gerektirdiğinden hataları artıracağı düşünülerek uygun görülmemiştir. Örneğin bazı bölümler, sayısal tabanlı; bazıları sözel tabanlı; bazıları ise yabancı dil tabanlı öğrenci aldığı gözardı edildiğinde hata yapmak olasıdır. Bu nedenle araştırmada, bölümler bazında kendi içinde en fazla benzerliğe sahip fakülte olarak İİBF öğrenci kitlesi, tercih edilmiştir.

İİBF, Muğla Üniversitesinin kuruluş tarihi olan 1992'den önce Dokuz Eylül Üniversitesi bünyesinde, 1970'li yıllara dayanan köklü bir geçmişe sahiptir. Mezun ettiği ve şu anda bulunan öğrenci sayısı itibariyle de üniversitenin toplam öğrenci sayısı içerisinde büyük bir yekun oluşturmaktadır.

Sonuç olarak bu çalışmada; öğrencilere ait geçmiş verilere dayalı olarak bilgi keşfi gerçekleştirmek istenmektedir. Bu görevi yerine getirebilmek için Muğla Üniversitesi Öğrenci İşleri ve Bilgi İşlem Dairesi başkanlıklarının izinleri ile İİBF öğrencilerine ait demografik ve eğitim verileri alınmıştır. Alınan bu veriler daha sonra SQL sunucuda filtrelenerek çalışmanın daha sağlıklı olabilmesi açısından İİBF'de mezun vermiş; iktisat, işletme ve kamu yönetimi bölümlerine ait 1995 yılı ve sonrası öğrenci verileri üzerinde yapılması uygun görülmüştür.

3.2.2. Verinin ihtiyaçlarına karar verme

Eldeki veri kümesi geniş çapta demografik veri ve eğitim süreci ile ilgili geçmiş verileri içermektedir. Bütün bunlar hangi sütun veya sütunların başarılı/başarısız kişilere ait deseni en iyi tanımladığı tahmini için model oluşturmakta kullanılacaktır.

Öğrencilerin eğitim başarısını elde bulunan veride iki sütun tanımlamaktadır: Mezuniyet durumu ve dönem sonu ortalama notu. Hedef sütun olarak bunların kullanılması uygundur. Okulda kalınan süre de bir gösterge olabilir ancak burada kalınan sürede başka faktörler (dikey geçiş yapanlar, alınan cezalar, kayıt dondurmalar veya ayrılmalar) de etkili olmaktadır. Dolayısı ile yine mezuniyet durumu bilgisine ihtiyaç duyulmakta ve birlikte sorgulanma durumu ortaya çıkmaktadır

3.2.3. Kullanılacak analiz türüne karar verme

Öğrenci sınıflandırma analizi önceden tanımlı grup (başarılı veya başarısız) için her bir öğrenciye ait olan geçmiş karakteristikleri tanımlamayı amaçlar. Veri madenciliğinde bir teknik olan karar ağacı sınıflandırması, tarihi veritabanı ve endüktif bir şekilde bulunan veriyi inceleyerek sınıflandırma modelleri oluşturur. Bu desen, şu anda varolan veriyi anlamak ve yeni gelenlerin nasıl davranacaklarını kestirmekte kullanılabilir.

Sınıflandırma, yeni bir nesnenin özelliklerini incelendikten sonra onu önceden tanımlanmış sınıflardan birine atama işlemidir. Ağaç yapısında kuralları üretmek kolaydır. Anlaşılması kolay modeller sağlar. Az bir veri hazırlığı ile işlemler gerçekleştirilebilir. Bu nedenle kolay bir şekilde bilgi teknolojisi ile bütünleşebilir ve yüksek seviyede otomasyon sağlar. Karar ağaçları bu özellikleri nedeniyle en popüler sınıflandırma tekniğidir (Berry ve Linoff, 2000).

Karar ağaçları, girişler ne kadar karmaşık olursa olsun anlaşılır kurallar üretir. Genellikle ağaç boyunca herhangi bir yolu izlemek kolaydır. Bu nedenle yol boyunca kararları açıklamak kolay olur. Her bir ayırma için gerçekleştirilen hesaplamalar kaynak tüketici değildir. Uygulamada, algoritmalar karar ağaçlarını düşük dallanma faktörü ile her bir düğümde basit denemelerle üretmeye meyillidir. Dolayısı ile ağaç kontrol dışı büyümez. Karar ağaçlarını kullanılarak, alanlardan (sütun) hangisinin eğitim kayıtlarının bölümlenmesi işleminde en iyi olduğu belirlenebilir. Bu kullanıcıya hangi değişkenin verileri en çok etkilediğini anlama imkanı sağlar. (Güvenç, 2001).

Bu çalışmada öğrenci gruplarını kendi özellik değerleri ile sınıflandırmak amaçlandığından ve oluşturulan modelin anlaşılırlığına daha uygun olarak görüldüğünden bir sınıflandırma algoritması olarak kullanılan karar ağaçları tercih

edilmiştir. Bu amaçla veritabanları üzerinde model oluşturmayı sağlayan Analiz Hizmetlerinde (Microsoft SQL Server 2000 Analysis Services SP3) hazır olarak bulunan MDT[41] (Microsoft Decision Trees) kullanılmıştır. Soni vd., (2002), MDT'nin giriş niteliklerinin sayısı ve içerdiği durum sayılarındaki artışlara göre eğitim süresindeki değişim gibi birçok performans analizini gerçekleştirerek sonuçları göstermektedir. MDT, olasılıksal (probabilistic) bir sınıflandırma ağacıdır ve aslen C4.5 algoritmasının benzeridir. Fark olarak ayırma kriteri olarak entropi (entropy) değeri yerine varsayılan olarak bayes puanlaması (bayesian score) kullanılmaktadır (Soni vd., 2002). C4.5 ise ID3 karar ağacı algoritmasının bazı eklentiler ile geliştirilmiş halidir. Şu anda See5/C5.0 adında ticari bir program olarak pazarlanmaktadır.

3.2.3.1. Karar ağacı ile sınıflandırma

Bir karar ağacı, yapısında bulunan düğüm, dal ve yaprakları ile tahminleyici bir model olarak görülür. Herhangi bir karar düğümü gerçekleştirilen denemeyi belirtir. Bu denemenin sonuçları herhangi bir veri kaybı olmaksızın ağacın dallara bölünmesine neden olur. Bölünme kararı o anda düğümde yapılır ve bir daha geri dönülmez. Aynı zamanda tek değişkenlidir. Tüm bölünmeler sıralı olarak yapılır, dolayısıyla her bir bölünme kendisinden bir öncekine bağımlıdır. Sonuç olarak tüm gelecek düğümler ilk bölünmeye bağımlıdır. Bunun anlamı ilk bölünme farklı seçilirse son çözüm çok değişik olabilir. Ağacın her bir dalı sınıflandırma sorusunun cevabı olabilir aksi halde diğer karar düğümüne veya ağacın en altına, yaprak düğüme, yollayacaktır. Yapraklar veri kümesi sınıflandırmasının kısımlarıdır. Karar ağacı süreci kök düğüm ile başlar ve yaprak düğüme ulaşıncaya kadar her bir izleyen düğüme doğru hareket eder.

Karar ağaçları bazı önemli bilgi parçalarını tahmin edebilmek için orijinal veri kümesinde ayrımlar oluşturur. Ağacın her bir yaprağı bir ayrımı gösterir. Tahminleyici ayrımlar, tahmin edilen bilgi ve tahminleyici ayrımı tanımlayan karakteristik açıklama açısından benzerdir. Sonuç olarak, karar ağaçları ve algoritmalar karmaşık olabilir; ancak çıkan sonuçların anlaşılması kolaydır.

---

[41] Microsoft karar ağacı algoritması

Karar ağaçları sayısal olmayan verileri işlemede oldukça iyidir. Kategorik veriyi kabul etmekteki bu kabiliyeti, veri dönüşümleri ve sinirsel ağların aslında olan tahmin edici değişkenlerdeki patlamayı azaltmaktadır. Bazı karar ağaçları sürekli değişkenlere destek vermez. (regresyon ağacı kurulamaz), bu durumlarda eğitim kümesindeki bu değişkenler çıkış sınıflarında depolanmak zorundadır.

Karar ağaçları veri sınıflandırmada güçlü olmasına karşın, kestirim (estimation) açısından uygun değildir. Ve yine eğilimleri görülebilir yapacak şekilde veri sunumunda yeterli çaba sarf edilmez ise zaman serisi verileri içeren problemlerde karar ağaçlarını kullanmak oldukça zordur.

3.2.3.2. Karar ağacı metodolojisi ve ilgili ölçümler

Karar ağacı sınıflandırması yöntemi, örnek kümeler kullanılarak gerçekleştirildiğinden denetimli öğrenime girmektedir[42]. Ağacın kalitesi sınıflandırmanın doğruluğu ve ağacın boyutuna göre değişmektedir. Yöntem, öncelikle eğitim örneklerinin bir alt kümesini seçer. Bu işlemler doğru karar kümesi bulununcaya kadar devam eder. En son çıktı, her bir yaprak bir sınıf adını taşıyan, her bir iç düğüm bir niteliği belirten bir ağaçtır.

Karar ağaçları özyinelemeli kısımlara ayrışım (recursive partitioning) olarak bilinen yöntem ile oluşturulur (Berson ve Smith, 1997). Bu yinelemeli olarak veriyi kısımlara ayırma işlemidir. Algoritma, bütün veri üzerinde uygun bir şekilde çalışan mümkün olan en iyi ağacı oluşturmak için arama gerçekleştirir. İşlem, önceden sınıflandırılmış kayıtları içeren bir eğitim kümesi ile başlar. Sınıflara ayıran ağacı inşa edebilmek için ağacın her dal noktasında sorulacak mümkün olan en iyi soru bulunmalıdır. Hedef, tahmin değerine göre olabildiği kadar homojen ağaç yapraklarına sahip olmaktır. Farklılık (diversity) ölçümü iki kısım için hesaplanır ve en iyi ayırma, farklılıkta en büyük azalma gösterebilendir. Ağaç kesin bir boyuta büyültüldükten sonra, algoritma modelin verilerin üstüne uyup uymadığını görmelidir. Bu da çapraz değerlendirme yaklaşımı (cross validation approach) tarafından yapılır.

---

[42] Denetimli öğrenme ile ilgili daha detaylı bilgi Bölüm 2.2.3'te verilmiştir.

Ağacın boyutu durdurma kuralları (stopping rules) ile kontrol edilebilir. Genel durdurma kuralı, basitçe ağacın büyümesi istenen maksimum derinliğe sınır koymaktır. Bu sayede kök ile en dış yaprak arası mesafe sınırlanmış olacaktır. Bir diğer durdurma kuralı ise düğümlerde kayıt sayısı için alt sınır koymaktır. Böylece bu sınırın altında ayrım gerçekleşmeyecektir. Budama (pruning), durdurma kurallarına alternatif olarak kullanılabilmektedir. Burada ağacın büyümesine bir sınır getirilmez. Sonra, yerleşik bir arabuluculuk sistemi ile ağaç geriye doğru doğruluğa uzlaşmayan en küçük boyuta gelinceye kadar budanır yani sadeleştirilir (Winston, 1992).

Bir sistemdeki düzensizlik seviyesini gösteren entropi, bilgi teorisinde sıkça kullanılan bir ölçüdür. Bir niteliğin entropi değerini yükseltmek, belirsizliği artırır. Sonuç olarak, ağaç boyunca nitelikler entropi değerinin artış sırasıyla seçilir. Böylece ağacın kök düğümünü ayıran nitelik en düşük entropi değerine sahip olmaktadır (Jackson, 1990; Ignizio, 1991). Eşitlik 4.1'de verilen bir niteliğe ait entropi değeri bulunmaktadır.

$$H(C \mid A_k) = \sum_{j=1}^{M_k} p(a_{k,j}) \times \left[- \sum_{i=1}^{N} p(c_i \mid a_{k,j}) \log p(c_i \mid a_{k,j}) \right] \qquad (4.1)$$

burada,

$H(C|A_k)$ Niteliğin sınıflandırma özelliğinin entropi değeri A_k
$p(a_{k,j})$= j değerindeki k niteliğinin olasılığı
$p(c_i|a_{k,j})$= k niteliğinin j'inci değerinde iken sınıf değeri c_i'nin olasılığı
M_k= A_k nitelik değerlerinin sayısı; j=1,2,....,M_k
N= farklı sınıf sayısı; i=1,2,...,N
K= nitelik sayısı; k=1,2,....,K

Köşeli parantez içinde bulunan terim bilgi olarak adlandırılır. Dolayısıyla entropi, olası çıkışlardaki bilgilerin toplamı ile bunların olasılığının çarpımıdır. Logaritmada taban olarak genelde 2 alınır. Bu nedenle bilgi bitler ile ölçülür.

Bir G kayıt kümesi kategorik nitelik üzerinde P1,P2....,Pn sınıflarına ayrılırsa, sınıf elemanı G yi tanımlamak için gerekli olan bilgi, Eşitlik 4.2'deki gibi gösterilir.

$$\text{Bilgi (G)} = -(p_1 \log (p_1) + p_2 \log (p_2) + \ldots . p_n \log (p_n)) \qquad (4.2)$$

burada pi, Pi sınıfına ait olasılık dağılımıdır. Eğer G kategorik olmayan G1,G2,...Gn kümelerindeki Y niteliğinde ilk ayrım ise, bir elemanı tanımlamak için ihtiyaç duyulan bilgi;

$$Bilgi(Y,G) = \sum_{i=1}^{n} \frac{|G_i|}{|G|} \times Bilgi(G_i) \qquad \textbf{(4.3)}$$

Bir çok sistem bilgi kazancını kullanmaktadır. Bilgi kazancı, yani orijinal bölümün entropi değeri ile sonuç ayrım bölmelerinin toplu entropi değeri arasındaki farktır. Bunlara en fazla bilgiyi kazanan testi seçerler diğer bir deyişle düzensizliği en aza indirirler. Bu durum Eşitlik 4.4'te verilmiştir.

$$\text{Kazanç (Y,G)} = \text{Bilgi (G)} - \text{Bilgi (Y,G)} \qquad \textbf{(4.4)}$$

Eğer bir niteliğe ait çok sayıda değer var ise, (4.5)'te gösterilen bilgi kazanç oranı tercih edilir.

$$\text{KazançOranı} = \frac{\text{Kazanç(G,Z)}}{\text{AyrımBilgisi(G,Z)}} \qquad \textbf{(4.5)}$$

burada Z, G kümesindeki her bir kayır için farklı değere sahip bir niteliği ifade eder. AyrımBilgisi(Z,G), kategorik nitelik Z'nin değeri üzerinde G'nin ayrımında oluşan bilgi olarak tanımlanır.

$$AyrimBilgisi(Z,G) = I\left(|G_1|/|G|, |G_2|/|G|, \ldots\ldots, |G_m|/|G|\right) \qquad \textbf{(4.6)}$$

burada {G1,G2,...Gm} Z değerinin sebep olduğu G'nin parçalarıdır. Bunlardan başka bir çok diğer değerlendirme fonksiyonları bulunmaktadır. Gini dizini (Gini Index), Ki Kare test (chi square test) bunlardan bazılarıdır.

3.2.3.3. Karar ağacı algoritmaları

Karar ağacı algoritmalarında ağaç oluşturulurken gerçekleştirilen işlemler birbirlerine çok benzemektedir. Bu algoritmalar, veri kümesini hedef değere göre özdeş durumlarda parçalara ayıran tüm olası ayırt edici nitelik değerlerine göre taramaktadırlar.

ID3,C4.5 (Quinlan, 1993), CART (Classification and Regression Trees; Breiman vd., 1984) ve CHAID (Chi-squared Automatic Interaction Detection; Kass, 1980) gibi bir çok çeşit algoritma ile karar ağaçları oluşturulabilmektedir. CART ve CHAID'de yeni ölçütler (metrics) kullanılmaktadır. CART ile farklı olarak sadece ikili ağaçlar (binary tree) oluşturabilmektedir. C4.5, bir düğüme karşılık daha çok sayıda dal üretmesi dışında CART'a benzemektedir C4.5 ve CART ile hem kategorik hem de sürekli değerler işlenebilirken; CHAID, sadece kategorik değişkenleri kullanabilir. Bu nedenle CHAID kullanıldığı zaman süreklilik gösteren değişkenler gruplanmalıdır. C4.5 ve CART'dan farklı olarak CHAID ağaç oluşturma işleminde budama (pruning) yapar. Bu nedenle sonuçlarda birebir uygunluk gerçekleşmez (Berson ve Smith, 1997).

Geliştirilen bu algoritmalar içerisinde ID3, 1970'lerde tanıtımı yapılan en eskilerinden birisidir. ID3 tahmin edicilerinin seçerken ve onların ayrılma değerlerini belirlerken, ayırmanın sağladığı bilginin kazancı baz alınır. Kazanç, ayrımlar gerçekleştirilmeden önce ve sonrasında doğru tahmin edebilmek için gerekli olan bilginin miktarını temsil eder ve orijinal parçadaki entropi değeri ile sonuçtaki ayrımlarda elde edilen parçalardaki toplam entropi değerindeki farkı tanımlar. C4.5, ID3'nin geliştirilmiş şeklidir ve burada budama bulunmaktadır.

Aşağıda çalışmada kullanılan karar ağacı algoritmasının da temelini oluşturan ID3 algoritması tanıtılmaktadır.

3.2.3.4. Quinlan ID3 algoritması

Tipik bir karar ağacı öğrenme sistemi olan ID3 (Quinlan, 1986), yukarıdan aşağıya geri çevrilemez bir stratejiyi benimsemektedir ve araştırma uzayının sadece belirli bir parçasını arar. Basit bir ağaç bulunacağını garantiler. ID3, bir nesneyi sınıflandırmak

için beklenen test sayılarını en aza indirme niyetiyle bilgi-teorik[43] tabanlı yaklaşımı kullanmaktadır. ID3'ün nitelik seçim kısmı, karar ağacının karmaşıklığının bu mesaj tarafından taşınan bilginin miktarı ile kuvvetli bir şekilde ilişkili olduğu varsayımına dayanmaktadır. Bilgiye dayanarak sezgisel yaklaşım ile (heuristic) en yüksek bilgi kazancı sağlayan niteliği seçer. Nitelik elemanların sınıflandırmak için sonuç alt ağaçlarında ihtiyaç duyulan bilgiyi en aza indirger.

ID3 sistemi, sınıflandırmada değerlendirme fonksiyonu olarak Eşitlik 4.4'te verilen bilgi kazancını kullanmaktadır. ID3 algoritması kategorik olan ve olmayan nitelikler kümelerinde kullanılabildiği gibi, kategorik niteliklerde ve aşağıda verilen algoritmada da görüldüğü gibi eğitim kümesinde kullanılabilmektedir (Ingargiola, 1997).

```
Fonksiyon ID3 (R:kategorik olmayan nitelikler kümesi,
               C:kategorik nitelik,
               S:eğitim kümesi) karar ağacı getirir;
Başla

      Eğer S boş değer ise, başarısız değeri ile tek düğüm getir;

      Eğer S kategorik nitelik olarak tüm kayıtlar için aynı
      değeri içeriyor ise, bu değer ile bir düğüm getir;

      Eğer R boş değerde ise, S'nin kayıtlarında bulunan
      kategorik nitelik değerlerinde en sık gözlenen değer ile
      bir düğüm getir;

            D, R'nin nitelikleri arasında en büyük kazançta
            olsun;
            D niteliğinin değerleri {dj|j=1,2,…,m}olsun;
                  {Sj|j=1,2,…,m} S'nin alt kümeleri olsun ve D
                  niteliği için sırasıyla dj değerinde kayıtları
                  içersin;
            D isimli kök ve sırasıyla d1,d2,…,dm isimli ayrımlı
            diğer sınıfları ile ağacı getir;

      ID3 (R-{D},C,S1), ID3 (R-{D},C,S2),……, ID3 (R-{D},C,Sm);

Bitir ID3;
```

[43] Knowledge-theory

3.2.4. Kullanılan denetleme ölçüsü

Modelin etkinliğini belirlemede yaygın olarak kullanılan kaldıraç grafiği (lift chart), yöntemi tercih edilmiştir. Kaldıraç grafiğinin tanıtımı ikinci bölümde (Bölüm 2.2.4) verilmiştir. Kaldıraç grafiği oluşturabilmek için test veri kümesi üzerinde Ek4'te gösterilen tahmin sorgusu oluşturulmuştur. Sonra sonuçlar veri kümesindeki bilinen değerler ile karşılaştırılmıştır. Test veri kümesi hedef sütunları içerdiği için bu mümkün olmaktadır.

3.3. Veritabanı İşlemleri

Üniversite öğrencilerini tanıma ve onları eğitimde başarıya götüren sebepleri araştıran çalışmanın tümü gerçek öğrenci verileri ile SQL sunucu ile yönetilen "STUDENT" adlı, yerel bir veritabanı üzerinde gerçekleştirilmiştir.

3.3.1. Muğla Üniversitesi öğrenci verilerini tutan veritabanının hazırlanması

Üniversite veritabanından alınan öğrenci verileri çok sayıda tablodan oluşmaktadır. Her bir tablo belirli özellik değerlerini barındıran sütunlardan meydana gelmiştir. MÜKÜP, veriler üzerinde model oluştururken tek bir tablo ile çalışabilmektedir. Bu nedenle model oluşturmada ilgili olduğu düşünülen 13 tablo seçilip Ek2'de gösterildiği gibi birbiriyle ilişkilendirilerek ve ilgilenilen kitleyi içerecek şekilde filtrelenerek aynı veritabanı üzerinde "VIEW_STUDENTPROFILE" adlı bir görünüm oluşturulmuştur. Görünümler, gerçekte çok sayıda tablodan oluşan veritabanı üzerinde tek bir tablo gibi çalışabilme izni verebilmektedir. Tüm işlemler bu görünüm üzerinden devam etmiştir. Burada toplam 111 sütun bulunmaktadır.

3.3.2. Analiz sunucusunun (Analysis Server) hazırlanması

VM modeli oluşturmak ve bunları çalıştırabilmek için analiz sunucusu üzerinde yeni bir veritabanı oluşturmak gereklidir. Bu yeni veritabanı içinde de VM modelleri oluşturabilmek için bir veri kaynağı oluşturulur. Veri kaynağı oluşturulurken sağlayıcı, sunucu, güvenlik denetim şekli ve veritabanı belirtilmelidir. Örneğin bu çalışmada,

sağlayıcı olarak **Microsoft OLE DB for SQL Server**; sunucu olarak çalışmanın kurulu olduğu sunucu adı olan **dmserver**; **Windows NT Integrated security** güvenlik denetimi ve yine üzerinde çalışılan **student** veritabanı seçimi yapılmıştır. Standart olarak **dmserver-student** adlı veri kaynağı oluşturulur. Bu isim, veri kaynağının farklı isimli bir kopyası alınma şeklinde değiştirilebilmektedir.

3.3.3. SQL Sunucu veritabanına bağlantı kurma bölümü

Veriler veritabanında tam anlamıyla saklandıktan sonra, MÜKÜP ile veritabanı arasında bağlantı kurmak gerekmektedir. Ek1-Form1'den de görüldüğü gibi, MÜKÜP kullanıcı ara yüzünde veritabanı bağlantısı kurabilmek için ara yüz üzerinde özel bir alan bulunmaktadır. Burada veritabanının bulunduğu sunucunun ve veritabanının ismi girildikten sonra *Bağlantı Kur* düğmesine basmak yeterlidir. Bağlantı kurulana kadar ara yüzde bulunan diğer tüm düğmeler işlem yürütülemeyeceği için pasif durumdadır.

3.3.4. Tablo seçimi

MÜKÜP'te *Tablo Yönetimi* seçildiğinde, Ek1-Form2a'da gösterilen *Tablo Yönetimi* formu açılır. Burada;

- Bütün bir tabloyu kopyalama,
- Bir görünümü (view) tabloya çevirme,
- Belirli sütunlar seçilerek bunları yeni bir tabloya yerleştirme,
- Veritabanından seçilen tabloyu bırakma,
- Mevcut bir tablonun daha az sayıda kayıt içeren bir örneğini oluşturma (Ek1-Form2b'de Örnekleme formu gösterilmektedir. Bu form, Tablo Yönetimi formundan açılır.), işlemleri gerçekleştirilebilmektedir. VM'ye tabi tutulan veriye göre yukarıda listelenen tablo yönetim teknikleri tercih edilebilir.

3.4. Veri Hazırlama Bölümü

Veritabanı ve kullanılacak tablo seçimi yapıldıktan sonra kullanılacak tablonun modele uygun hale getirmek için bir dizi işlemden geçirilir. Veri temizleme, veri dönüştürme, veri inceleme ve bunlara ait alt görevler, bir bütün olarak verinin

hazırlanması safhasıdır. Bu üç görev model oluşturulmadan önce gerçekleştirilir ve birbirleri ile ilişkilidir.

3.4.1. Veri temizleme işlemi

Bu bölümde, aşağıda belirtilen üç probleme yönelik çözüm teknikleri kullanılmıştır.

- Çok sayıda boş değer içeren sütunlar için; Model oluşturma sürecinden dışlanıp dışlanmayacak niteliklere karar verebilmek için her bir nitelik için boş değer yüzdesi hesaplanır.
- Çok az farklı veya çok fazla farklı değerler alan sütunlar için; her bir niteliğin ortalaması, minimum, maksimum ve farklı değer sayısı hesaplanır. Bu bilgi kullanışsız gözüken sütunları dışlamakta kullanılır.
- Sütunun normal dağılımının dışına düşen kayıtlar için; sınır dışı değerler hesaplanır ve içinde yerleştiği satırlar işaretlenir. Bu saturlar geliştirilen çözüme uygun olacak şekilde çıkarılabilir veya hücre değerleri sütunun ortalama değeri ile değiştirilebilir.

Bu problemlere yönelik MÜKÜP Temizleme formunda üç sekme yer almaktadır (Ek1-Form3a,b,c). Temizleme formuna MÜKÜP kullanıcı ara yüzünde *Veri Hazırlama* bölümünde bulunan *Temizle* düğmesi seçilerek ulaşılabilir.

3.4.1.1. Sütunlardaki boş değer yüzdelerini hesaplama

Temizleme formunun ilk sekmesi varsayılan olarak *Boş değer Yüzdesi*dir. Sütun içerisindeki boş değer yüzdesini hesaplama bu üç görev içinde en az zaman alanıdır. Bu nedenle ilk adım olarak tercih edilmiştir.

Veri kümesi hakkında belki de ilk dikkati çeken çok sayıda sütuna sahip olmasıdır. Pek çok sayıda sütuna sahip olma, veri ile çalışmaya başlarken aşırı derecede dağınıklık meydana getirmektedir. Bu nedenle ne kadar gereksiz sütunu çıkarılabilirse ilerideki görevleri yerine getirirken hesaplama için o kadar az zaman harcanılır. Ayrıca çok sayıda boş değer içeren sütunlar; sonuca bir etkileri olmazken, modelin doğruluğunu da ters yönde etkilerler. Bu nedenle çözüm geliştirilirken çok fazla boş değer içeren sütunlar modelde kullanılacak tablodan başlangıçta çıkarılması uygundur. Çıkarılan

sütunlar daha sonra Ek1-Form3a'da gösterilen form içinde yer alan FlexGrid denetiminde ekrana gelmekte ve SQL sunucuda ayrı bir tablo içersinde bu sütunların isimleri tutulmaktadır. Ancak asıl tablodaki veriler kaybetmemek istenilirse *Tablo Yönetimi* formundan farklı bir isimde tablo oluşturularak bu işlemler gerçekleştirilebilir.

Bu temizleme görevini yerine getirmek için, üç değer kullanılır:

- Yüzde kesim değeri –hangi yüzde değerinin üstünde boş değer barındıran sütunlar tablolar çıkarılacağını belirler-
- Tablodaki satır sayısı
- Tablodaki her bir sütun için boş değer sayısı

Burada iş, istemci ve sunucu arasında ayrılabilir. Bu durumda sütun verisi, sunucudan alınıp istemciye yerleştirilir, teste tabi tutulur ardından sütun kritere uyuyor ise sütunu tablodan çıkarmak için sunucuya dönülür. Bu tür çözümde, çok sayıda sunucu istemci arasında ileri-geri gitme dikkati çekmektedir. Çalışılan tabloda bulunan sütun sayısı yüze yakındır. Dolayısıyla bu döngü sayısı dikkat çekici oranda artacaktır. Alternatif olarak, bütün işi sunucu tarafında gerçekleştirmek; yani program kodumuz tarafından çağrılan saklı yordamlar[44] oluşturmaktır. Bu yöntem diğeri ile kıyaslandığında 20 kat daha hızlıdır. Bu nedenle MÜKÜP'te bu ikinci teknik kullanılmıştır.

3.4.1.2. Sütun özelliklerini hesaplama

- Veri temizleme formunun ikinci sekmesi, özellik hesaplamalarıdır (Ek1-Form3b). Her bir sütuna ait belirli özellikleri hesaplar. Burada üç görev gerçekleştirilebilmektedir:
  - Özellikleri hesaplama,
  - Daha önceden hesaplanmış özellik tablosunu gösterme,
  - Tablodaki özellik bilgisine dayanılarak sütunu tablodan çıkarma.

Sayısal sütunlarda hesaplanması kolay ve veri hakkında fikir verebilen bir kaç sütun özelliği vardır. Bunlar;

---

[44] Stored procedures

- Minimum değer,
- Maksimum değer,
- Ortalama,
- Standart sapma,
- Sütundaki farklı değerlerin sayısıdır.

İki nedenle bu değerler hesaplanmaktadır:

- Veri kümesinin her bir sütunundaki kayıtların dağılımını en iyi şekilde anlamak.
- Veriyi inceleme adımında değişkenler için gerekli değerleri hesaplamak.

Özellik hesaplamak için program, tablodaki her bir sayısal sütun içerisinde yinelemeli olarak dönen bir saklı yordam çağırır. Hesaplanan sütun özellikleri, SQL Sunucuda bir tabloda saklanır. Çalışmanın sonunda sonuçlar form üzerinde tablo şeklinde (özellik tablosu) görüntülenir. Görüntülemede Visual Basic araç kutusunda[45] yer alan *FlexGrid* denetimi[46] kullanılmıştır.

Burada asıl istenen tablodaki sütun sayısını azaltmaktır. Kullanılan veriler iyi tanınır ise, model kurmaya gelindiğinde daha doğru tercihler yapılabilir. Hangi sütunun çıkarılacağına karar verebilmek için birkaç değişik değere bakılır. Örneğin bir sütunun aldığı farklı değer sayısına bakılarak; çok fazla sayıda farklı değere sahip ise modele bir katkı sağlamayacağı düşünülerek çıkarılabilir. Anahtar sütun, her bir kaydı tanımladığı için bu kuralın dışında tutulur. Yine bazı evet-hayır gibi iki değer alan özel sütunlar haricinde çok az farklı değer alan sütunlar yine modele katkıda bulunamayacağına karar verilirse tablodan çıkarılabilir. Tek farklı değer alan sütunlar zaten modele bir katkı sağlayamaz.

3.4.1.3. Aykırı değerleri işaretleme

Aykırı değer işaretleme, son temizleme görevi sekmesidir. Bu sekme, hangi değerlerin aykırı değer olarak işaretleneceğini belirlemekte kullanılır. Buraya kadar veri üzerinde sütun bazında temizleme işlemleri yapılmıştır. Gereksiz görülen bir sütun

---

[45] Toolbox
[46] Control

içerisindeki tüm veriler çıkarılmıştır. Burada ise veriler üzerinde odaklama hücreler seviyesindedir. Veri kümesi çok geniş olduğu için her bir hücreyi fiziksel olarak inceleyebilmek imkansızdır. Bu nedenle MÜKÜP'te, bu işlem otomatikleştirilerek öncelikle her bir sütundaki aykırı değerler bulunur. Daha sonra bu sekmede aykırı değer içeren satırlar, aykırı hücre değerleri kırmızı renkte vurgulanılarak görüntülenir. Ek1-Form3c, *Aykırı Değer İşaretleme* sekmesini göstermektedir. Burada vurgulanan hücreler gözden geçirilerek istendiğinde bu hücreyi tutan satır silinebilir veya hücre değeri sütunun ortalama değeri ile değiştirilebilir.

Aykırı değerleri işaretlemeden önce üç değişkeni belirlemek gerekmektedir:

Standart Sapma Cinsinden Uzaklık: Bir değer aykırı olarak dikkate alındıktan sonra ortalamadan standart sapma cinsinden uzaklık.

Maksimum Oran: Maksimum değerin gerçekten dışarıda olduğunu belirten orandır.

Yüzde Kesim (cutoff) : Aykırı sınırını geçen değerlerin yüzdesi. Bunlar; gerçekten aykırı mı, değil mi karar vermek için kullanılır. Dağılım normal olarak farz edilemeyeceğinden çok fazla sayıda değer ortalamadan önemli uzaklıkta bulunabilir. Bu durum var ise bu tür değerler geçerli olabileceğinden işaretlenmezler.

Program algoritmasında aykırı değer işaretlemesinde aşağıda verilen formül kullanılmıştır.

Aykırı değer = Ortalama + Standart sapma * Standart sapma cinsinden uzaklık

Burada, öncelikle standart sapma cinsinden uzaklık belirlendikten sonra, değerler aykırı olarak dikkate alınabilir. Normal dağılıma göre, üç standart sapma yüzde 99.7 değerini kuşatmalıdır. Bu nedenle varsayılan değer 3'tür. Aykırı değer hesaplandıktan sonra, aşağıda gösterilen bir sonraki formül, veri kümesindeki maksimum değer ile aykırı değer arasındaki orana bakar.

Aykırı oranı = Maksimum değer / Aykırı değer

Eğer bu oran maksimum oran değerinden büyük ise, maksimum değer büyük olasılıkla aykırıdır. Eğer bu oran maksimum oran değerinden küçük ise, maksimum değer aykırı değildir.

Daha sonra aykırı değerlerin yüzdesi hesaplanır. Eğer bu değer, yüzde kesim değerinden yüksek ise bu değerleri aykırı değil olarak farz edilebilir. Fakat düşük ise bunlar aykırı olarak kabul edilebilir.

3.4.2. Veri dönüştürme işlemi

Veri temizlendikten yani problemli satır ve sütunlar çıkarıldıktan sonra, geride kalanlardan bazılarını istenen forma dönüştürme işlemi gerçekleştirilir. Bunu yapabilmek için "DTS Import/Export Wizard" sihirbazı kullanılmaktadır[47]. Burada *VBScript*[48] programlama dili kodları kullanılarak veriler dönüştürülür. Gerçekleştirilen dönüşümlere ait VBScript program kodları Ek3'te verilmiştir. Gerçekleştirilen iki çeşit dönüşüm vardır. İlk olarak giriş sütunları çok fazla sayıda duruma sahip ise, giriş sütunları durumlarının hedef sütununun durumlarını nasıl etkileyeceğini bulabilmek zordur. Bu problemi çözmenin yolu, giriş sütunlarındaki durumların sayısını düşürmektir. Sonsuz sayıda olasılıklara sahip olmaktansa belirli sayıda durumlar belirleyerek (durum1-durum2... gibi) sütun değerlerini bu uygun durum değerleri ile değiştirmek şeklinde uygulanmıştır. İkinci olarak eldeki sütunlarda hedef sütuna etki edebilecek veriler tek bir sütunda ve hazır bir formatta bulunmadığı durumlarda da dönüşüm kullanılabilir. Çalışmada bu tür bir dönüşüm olarak farklı iki sütun arasında gerçekleştirilen bir matematiksel işlem ile daha kullanışlı olabilecek yeni bir sütun elde edilmiştir. Gerçekleştirilen dönüşümlere ait daha somut bilgilere, Bölüm 4.2.3'ten ulaşılabilir.

MÜKÜP kullanıcı ara yüzünde *Dönüştür* düğmesine basıldığında DTS Import/Export sihirbazı açılır. Buradaki adımlar takip edilerek dönüştürme gerçekleştirilir ve kullanıcı ara yüzüne geri dönülür.

Bu aşamada, dönüşüme uğrayan sütunlardaki değerleri değiştirmek yerine dönüştürülen sütunları tabloya eklemek tercih edilmiştir. Böylece orijinal sütunlardaki değerler korunmuştur.

---

[47] (Başlat - Microsoft SQL Server - Import and Export Data) yolu izlenerek ulaşılabilir
[48] Microsoft tarafından geliştirilmiş Visual Basic dili tabanında script dili

3.4.3. Veri inceleme işlemi

VM modeli hazırlanırken, modelin etkinliğine yardımcı olan veya engel olan sütunlara karar verebilmek için veri incelemesi bölümünde görsel ve sayısal teknikler bir arada kullanılarak verilerin sütun boyunca nasıl dağıldığı ve farklı sütunların bir diğeri ile veya belirlenmiş hedef sütun ile nasıl ilişkide olduğuna bakılabilmektedir.

3.4.3.1. Grafiksel inceleme

Görsel teknikler, çok fazla sayıda sütuna hızlı bir şekilde bakma ve bunların arasındaki etkileşim hakkında genel bir fikir verirler. Ayrıca sayısal değer içeren sütunlar, korelasyon matrisinde olduğu gibi sayısal olarak incelenerek ilişkilerin kuvvetine toplu olarak bakılabilir.

MÜKÜP ile veri incelemesi grafiksel inceleme olarak histogram ve noktasal grafik; sayısal analiz olarak korelasyon matrisi kullanılmıştır (Ek1-Form5a veEk1-Form5b).

Karakter tabanlı sütunlar ile çalışırken grafiksel gösterim olarak sadece histogramlar kullanılmıştır. Veri doğal olarak gruplandığı için karakter tabanlı sütunlar için histogram, oluşturması en kolay olanıdır. Ekranda bu sonuçlar histogram olarak gösterilirken belirli sütunları kullanıp kullanmamaya karar vermekte yardımcı olabilecek bilgiler aranılmıştır.

Sayısal sütunların bazılarındaki verilerinin dağılımını keşfetmek için histogram kullanılmıştır. Sütunların özelliklerine bakıldığında bazı sütunların sadece bir veya iki farklı durum içerdiği görülmektedir. Burada bu sütunların daha iyi bir VM modeli oluşturmada yardımcı olup olamayacağına cevap aranılır. Bu sütunlardaki tüm popülasyon içerisinde hedef sütundaki pozitif ve negatif cevapların oranı benzer durumda ise yani tüm popülasyonda oran pek değişmiyor ise bu sütunların modelin etkinliliğine bir katkısı olamaz. Dolayısıyla tablodan çıkarılır.

Noktasal grafik, hedef sütunların durumlarının seçili sütuna karşı nasıl dağıldığını ifade eder. Bu dağılıma bakarak, VM modelini kurarken hangi sütunların dahil edileceği hakkında iyi bir fikir verebilir. Yalnız sayısal değer içeren sütunlar için kullanılabilir. Hedef sütunun durumları seçili sütuna karşı yansız bir şekilde yayılıyor ise, seçili

sütunun hedef sütun hakkında anlamlı bir bilgi vermediği; fakat hedef sütunun durumları gruplaşmış durumda ise seçili sütun, hedef sütunun sonucuna karar verirken kullanışlı olduğu söylenilebilir.

Grafik sekmesinin altında basit bir metodoloji izlenmektedir. Önce seçili sütun verisine dayalı olarak *Select* deyimi[49] oluşturulur. Daha sonra kayıt kümesi doldurulur ve *Microsoft Chart* denetiminin veri kaynağı ayarlanarak ekrana yansıtılır. Grafik türleri arasında (Histogram veya noktasal grafik) tek değişen, grafiğin biçimlendirilmesi; sütun türleri arasında değişen tek şey ise, Select deyiminin formülasyonudur.

3.4.3.2. Korelasyon matrisi

Korelasyon matrisi oluşturulmasında MÜKÜP, tüm matrisi (her bir sütuna karşı her bir sütun) veya kısaltılmış matris olarak adlandırılan sadece bir satırı (sadece hedef sütuna karşı her bir sütun) hesaplamak arasında seçenek sunmaktadır. Hedef sütunun hangisi olduğu zaten önceden belirlendiğinden, asıl önemli olan her bir sütunun hedef sütun ile nasıl ilişkide olduğudur. Ayrıca, VM modelinde bulunması gereken sütunları seçmemizde yardımcı olacak da bu durumdur. Ek1-Form5b'de gösterildiği gibi kısaltılmış matris, korelasyon matrisi sekmesinde hesaplanır ve gösterilir.

Bu rutin işlem, her yinelendiğinde değerlerde az bir farklılık oluşabilir. Bu beklenen bir durumdur; çünkü korelasyon hesaplamasında pratik olması açısından asıl tablo yerine asıl tablonun daha az kayıt taşıyan örneklemi kullanılmaktadır. Tüm matrise bakmak istenilirse daha fazla bir zaman ayrılmalıdır. Tüm matrisin hesaplanması ile sütunların sadece hedef sütun ile ilişkisi değil tablodaki diğer sütunlar ile olan ilişkilerinin derecesi görülebilmektedir.

3.5. Model Oluşturma Bölümü

Buraya kadar veri temizleme, dönüştürme ve inceleme işlemleri gerçekleştirilmiş ve veri hazır hale getirilmiştir. Bu aşamada VM modelini oluşturmadan bir önceki adım olarak asıl tablo ile ilgili iki şey yapılması gerekmektedir.

---

[49] SQL' de seçim işleminde kullanılan temel ifade

İlk olarak modelde hangi sütunların kullanılacağına karar verilip, model kurarken kullanılacak sütunları tutmak için yeni bir tablo oluşturmak gereklidir.

İkinci olarak tahminleme modellemesi yapabilmek için; aynı veri kümesinden alınmış eğitim (model kurma) ve test (denetleme) veri kümelerine sahip olmak gerekmektedir (Bölüm 2.2.3).

Yeni tablo oluşturabilmek için *Tablo Yönetimi* formu kullanılmıştır. Burada belirli sütunlar seçilerek yeni bir tabloya kopyalanabilmektedir. Yeni tablo oluşturabilmek için seçili tablonun sütunları seçilir ve farklı bir isme sahip yeni bir tabloya yerleştirilir.

3.5.1. Veri ayırma işlemi

Asıl tabloyu doğru bir şekilde ayırabilmek için yapılması gereken iki önemli görev bulunmaktadır. Her bir tablonun benzersizliğini sağlamak ve her bir tablonun satırlarını rasgele seçebilmek için satırları izlemek. Bu görevleri gerçekleştirmek için yine SQL sunucunun işlevselliklerinden faydalanılmıştır. İlk olarak hangi satırların eğitim, hangilerinin test tablosunda kullanacağını izlemek için ilgili formda ismi girilen iki yeni tablo oluşturulur. Bu kısımda asıl tablodan satırlar rasgele seçerek bu tablolara yerleştiren bir Transact-SQL deyimi kullanılmıştır.

Ardından program, hedef sütundaki pozitif ve negatif cevapların yüzdesini hesaplar ve Veri Ayırma formu üzerinde bu sonuçları görüntüler (Ek1-Form6). Gösterilen yüzdeler, yeni oluşturulan tabloların asıl tablodaki veriyi ne kadar iyi temsil ettiğini görmemizi sağlamaktadır.

Veriyi ayırma, kullanıcıya aynı kaynaktan türetilmiş verileri kullanarak model oluşturma ve bunu test etme izni vermektedir. Model oluşturulurken, *veri ayırma* bölümünde oluşturulan *eğitim tablosu* kullanılmaktadır.

3.5.2. Model oluşturma ve izleme işlemi

Ek1-Form7'de gösterilen *Model Oluşturma* formuna bakıldığında, kendi içinde üç bölüme ayrıldığı görülür. Birinci bölümde analiz sunucusu, veri kaynağı ve model ismi seçilir. Burada kurulan modelin türü ve verinin nereden geleceği tanımlanır. Modele sütunları ekleme ve çalıştırma işleminden önce gerçekte modele veri yerleştirilmemiştir.

Birinci bölümde veri kaynağını ayarlandığında sadece model, uygun veritabanına yönlendirmiş olur. Model oluşturma formunun ikinci bölümünde ise veritabanından tabloyu seçmek gerekmektedir. Sonra sütunların model tarafından nasıl kullanılacağı tanımlanır ve modele eklenir.

Model oluşturulurken seçilen sütunlar belirli parametre değerleri ile eklenmelidir. Bu çalışmada; seçilen sütunlar için sadece *veri tipi*, *kullanım* ve *içerik tipi* sütun özellikleri belirlenmiştir.

Veri tipi özelliği, sayısal veya karakter tabanlı gibi veri cinsini tanımlar. *Kullanım* özelliği, sütunun giriş, hedef sütun veya bu ikisi birlikte olduğunu belirtir. *İçerik tipi* özelliği ise sütundaki verinin sürekli veya kesikli olduğunu tanımlar. Hedef sütun model oluşturabilmemiz için kesikli olmak zorundadır. Kesikli değil ise hata meydana gelir. Ayrıca, anahtar sütunu, anahtar olduğu belirtilerek eklenmelidir

Boş bir karar ağacı kabuğu analiz sunucuda oluşturulur ve formdaki ağaç denetimine eklenir. Sütunlar, birer birer tabloya uygun bir biçimde parametrelenerek eklenir. Tüm sütunlar eklendikten sonra model çalıştırılabilir.

Birinci adımda model için sadece kabuk oluşturulur. Ancak form üzerindeki *çalıştır* düğmesi tuşlandığında, veriler kabuk boyunca geçirilir ve modeli tanımlayan ilişkiler oluşturulur.

Model çalıştırıldığında inceleme yapılabilir. Birinci model incelemesi yapıldıktan sonra veri kaynağı farklı tablolar kullanarak bu işlemler yinelenebilir. Farklı bir tablonun modelini oluşturmak için aynı analiz sunucusu fakat farklı kaynak tablosu ve model ismi kullanılır. Böylece birden fazla model oluşturulmuş olur.

Analiz Sunucusu[50], hem karar ağacı hem de kümeleme algoritmasını görüntülemeyi sağlamaktadır. Ancak örnek uygulama ile modelleri kuran analiz yöneticisi[51] arasında ileri-geri gidip gelmek, sonra modellere göz atmak ve sonra modelleri oluşturmak vs. gibi zaman alıcı ve takip edilmesi zor olan işlemler yerine form üzerinde ağaç gösterimi sağlayacak bir yol aranılmıştır. Çözüm olarak; daha kolay ve pratik bir şekilde modelleri

---

[50] Analisis server
[51] Analisis manager

görmek ve karşılaştırmak için üçüncü parti görüntüleyici olan *Angoss Tüketici Denetimleri* (Angoss Consumer Controls) tercih edilmiştir.

Model çalıştırıldıktan sonra yine *model oluşturma* formunda bulunan *incele* düğmesi seçilerek, görüntüleyici açılır ve yeni oluşturulmuş model görüntülenir. Diğer oluşturulan modelleri görebilmek için benzer işlemler yinelenir.

3.5.3. Model denetleme işlemi

Modeller kurulduktan sonra tahmin etme kabiliyetini test etmeden modeli kullanıma açmak yanlıştır. Bu çalışmada *Angoss SDK* tarafından sağlanan *kaldıraç grafiği göstericisi* kullanılarak bu durum gerçekleştirilmiştir.

Denetleme işlemini gerçekleştirebilmek için Ek1-Form8'de gösterilen *Denetleme* formu üzerinde bir model ve onun üzerinde yerleştiği analiz sunucusu, test verisi için kaynak ve bulmaya çalıştığımız hedef sütunun durumunu belirtmek gerekmektedir. Sonra modele göre test verisi için tahminleri oluşturan tahmin sorgusu oluşturulur ve sonuçlar kaldıraç grafiğine gönderilir. Tahminde kullanılan sorgu Ek4'te verilmiştir. Rasgele seçime nazaran model tarafından belirlenen hedef kitlenin başarı yüzdelerinin çizilmesi ile kaldıraç grafiği oluşturulmuş olur.

4. ARAŞTIRMA BULGULARI

Muğla Üniversitesi öğrencilerinin eğitimlerindeki başarısına etki eden kişisel özelliklerini keşfetme amaçlı bu çalışmada, ilgili olarak görülen 13 tablo ilişkilendirilerek tek bir tablo görünümüne çevrilmiştir. Bu tabloda mantıksal olarak ilgili olmayan giriş sütunları çıkarıldıktan sonra elde kalan sütunlar üzerinde veri temizleme işlemleri gerçekleştirilmiştir. Burada öğrencilerin dönem sonu başarı ortalamalarına dayalı olarak gruplanmış iki adet sütun hedef sütun olarak kullanılmıştır. Diğer sütunların hedef sütuna olan etkileri grafiksel ve istatistiksel analizler ile incelenerek, bu bağlamda anlamlı bulunan sütunlar ile karar ağacı modelleri oluşturulmuş ve bu oluşturulan modellerin, kaldıraç grafiği ile etkinlikleri denetlenmiştir.

4.1. Muğla Üniversitesi Öğrencilerine Ait Veritabanı Sorgulama Bulguları

Araştırma, 01.01.1995 tarihinden sonra kayıt yaptırmış Muğla Üniversitesi, İİBF fakültesine ait üç bölümdeki öğrencileri içermekte ve 4467 kişilik bir örneklem oluşturmaktadır. Bu öğrencilere ait bilgilerin dağılımları Tablo 4.1'de verilmiştir.

Tablo 4.1 Öğrenci bilgilerinin dağılımı*

DEĞİŞKEN		FREKANS (n)	YÜZDE (%)	DEĞİŞKEN		FREKANS (n)	YÜZDE (%)
BÖLÜM	İşletme	2234	50	YERLEŞİM BÖLGESİ	Ege	1519	34
	İktisat	1474	33		Marmara	1027	23
	Kamu Y.	759	17		Akdeniz	715	16
CİNSİYET	Erkek	3082	69		İç A.	536	12
	Kız	1385	31		Karadeniz	313	7
UYRUK	T.C.	4466	100		Doğu A.	223	5
	Diğer	1	0		G.Doğu A.	134	3
MEDENİ DURUM	Bekar	4463	100	LİSE OKUL TÜRÜ	Düz L.	3529	79
	Evli	4	0		Meslek L.	715	16
LİSE MEZUNİYET DERECESİ	Orta	2502	56		Özel L.	89	2
	İyi	1742	39		Anad. L.	89	2
	Pekiyi	223	5		İHL	45	1
HARÇ DURUMU	Alıyor	1340	30	ASKERLİK	Tecilli	2008	45
	Almıyor	3127	70		Bayan veya Yaşı küçük	2455	55
P. TÜRÜ	Eşit ağırlık	4467	100		Yapmış/Muaf	4	0

*Yüzde değerlerinde ondalık kısımlar yuvarlanmıştır.

Tablo 4.1 incelendiğinde fakültenin yarısını işletme bölümü öğrencileri, yaklaşık üçte birini iktisat ve kalan kısmını ise kamu yönetimi bölüm öğrencilerinin oluşturduğu görülmektedir. Öğrencilerin tamamına yakını T.C. vatandaşı ve bekar durumdadır ve erkekler kızlara göre sayıca baskın durumdadır. Öğrencilerin geldikleri yerleşim yerleri bölgelere göre gruplandığında, üniversitenin bulunduğu il ve buna yakın illerin dahil olduğu bölge öğrencilerinin ağırlıklı olduğu gözlenmektedir. Ancak burada Marmara bölgesinin Akdeniz bölgesinden yüksek değere sahip olması İstanbul, Kocaeli gibi nüfusça yoğun illerin Marmara bölgesine ait olmasından olabilir. Burada öğrencilerin üniversiteyi seçimlerinde memleketlerine olan yakınlığı gözettikleri yargısına varılabilir.

Öğrencilerin bitirdikleri liseler bazında düz liseler %79 gibi önemli bir yekun oluşturmaktadır. Bu da öğrencilerin eğitim geçmişlerinin büyük oranda benzediğini göstermektedir. Lise mezuniyet derecesine bakılırsa ağırlıklı olarak *orta* dereceli öğrencilerin bu bölümleri tercih ettiği görülmektedir. Öğrencilerin %30'u harç kredisi desteği almaktadır ve çok az bir kısmı askerliğini eğitimleri öncesinde yapmış durumdadır.

4.1.1. Öğrencilerin üniversiteye giriş puanları ve tercih sıraları

Tablo 4.2'de son iki yıla ait ÖSS puanlarına dair ilgili bilgiler verilmiştir. Son yılda tüm bölümlerin puanlarında yaklaşık 3 puanlık bir düşüş gerçekleşmiştir.

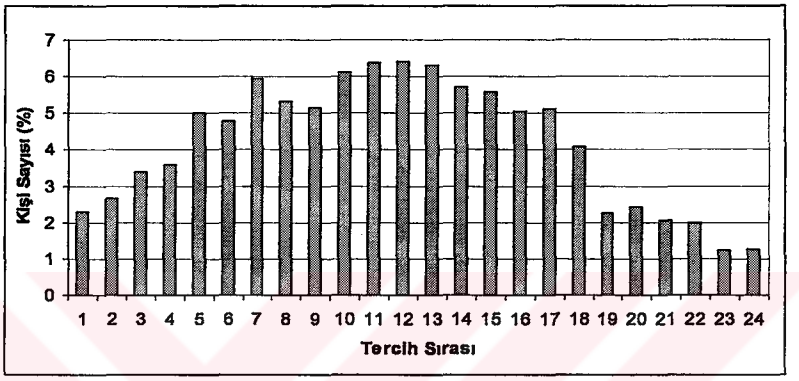
Tablo 4.2 Son iki yıla ait ÖSS giriş puanı istatistikleri

	2003			2004		
	İktisat	İşletme	Kamu Y.	İktisat	İşletme	Kamu Y.
Ortalama	227,99	230,91	227,40	225,51	227,43	223,36
Minimum	210,15	214,72	212,40	212,08	199,07	190,96
Maksimum	259,95	274,23	247,01	244,18	243,86	252,22
Standart Sapma	6,54	6,61	6,29	5,80	5,90	6,15

Tablo 4.3 kullanılarak elde edilen Şekil 4.1'de verilen grafik, öğrencilerin bölümlerini kaçıncı tercih ile kazandıkları dağılımını göstermektedir. Grafik normal bir dağılım göstermektedir. Öğrencilerin tercihlerinde bu bölümleri, genelde orta sıralarda yer aldığı gözlenmektedir.

Tablo 4.3 Üniversiteye girişte öğrencilerin tercih sırası verileri

Tercih	1	2	3	4	5	6	7	8	9	10	11	12
Kişi	75	87	112	118	165	157	196	175	170	201	210	211
Yüzde	2,28	2,64	3,40	3,58	5,01	4,76	5,95	5,31	5,16	6,10	6,37	6,40
Tercih	13	14	15	16	17	18	19	20	21	22	23	24
Kişi	208	188	184	166	168	134	74	80	68	66	41	42
Yüzde	6,31	5,70	5,58	5,04	5,10	4,07	2,25	2,43	2,06	2,00	1,24	1,27

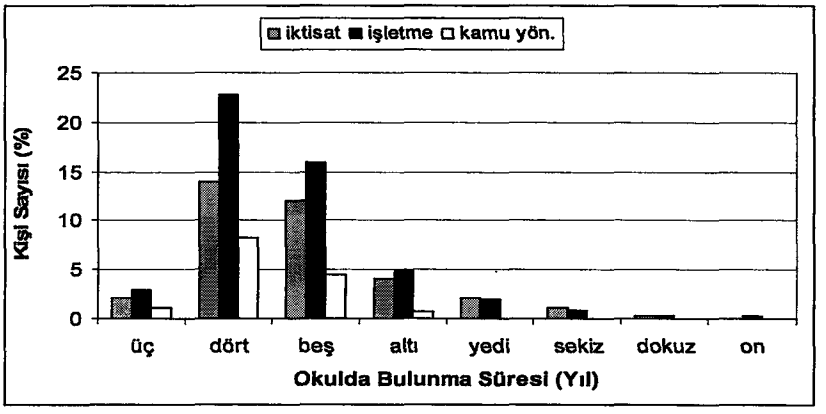


Şekil 4.1 Üniversiteye girişte öğrencilerin tercih sırası

4.1.2. Öğrencilerin üniversitede bulunma süreleri

Tablo 4.4 Mezun olan öğrencilerin okulda bulunma süresi verileri

YIL			üç	dört	beş	altı	yedi	sekiz	dokuz	on
BÖLÜM	İktisat	Sayı	47	322	279	93	47	26	8	1
		Yüzde	2,03	13,90	12,05	4,02	2,03	1,12	0,35	0,04
	İşletme	Sayı	69	527	370	114	44	18	8	5
		Yüzde	2,98	22,75	15,98	4,92	1,90	0,78	0,35	0,22
	Kamu yön.	Sayı	25	191	105	17	0	0	0	0
		Yüzde	1,08	8,25	4,53	0,73	0	0	0	0



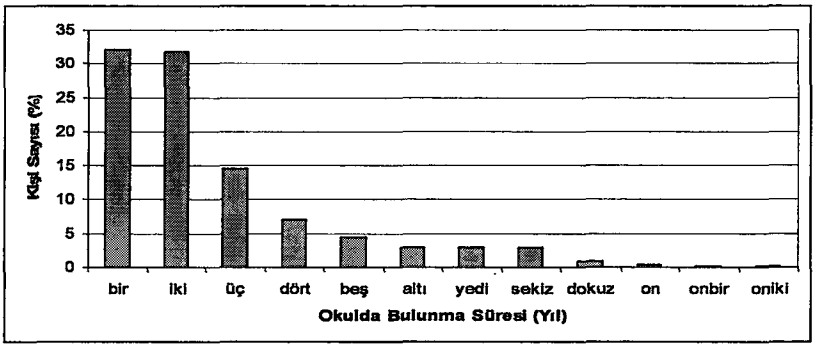
Şekil 4.2 Mezun olan öğrencilerin okulda bulunma süreleri

Şekil 4.2, Tablo 4.4'teki verileri kullanılarak oluşturulmuştur. Burada mezun olmuş öğrencilere ait okulda bulunma sürelerini gösterilmektedir. Bölümler bazında bir farklılık gözlenmemekle beraber, öğrencilerin çoğunluğu okulu normal sürede tamamlasa da belirgin bir şekilde bir yıllık uzatma gözlenmektedir. Okulu bitirme için verilen yedi yıl sürenin aşılması kayıt dondurma veya afla geri gelme gibi durumları içerir.

Şekil 4.3, Tablo 4.5'ten faydalanılarak oluşturulmuş olup; mezun olmadan okuldan ayrılan veya kaydı silinen öğrencilerin okulda bulunma sürelerini göstermektedir. İlk iki yıl içersinde kayıt silinmesi yüksek değerdedir. Beklentilerini farklı bir yer veya alanda karşılama isteğinin ilk yıllarda olması doğaldır. İkinci yıldaki ayrılmalar birinci yıla oldukça yakın olması dikkat çekicidir. Bunun bir sebebi, ÖSS sınavına ikinci yıldan sonra girişlerde katsayı kesintisinin olmaması olabilir.

Tablo 4.5 Mezun olamadan okuldan ayrılan öğrencilerin okulda bulunma süresi verileri

YIL	bir	iki	üç	dört	beş	altı	yedi	sekiz	dokuz	on	Onbir	oniki
Sayı	208	206	94	45	29	19	19	19	6	2	1	1
Yüzde	32,05	31,74	14,48	6,93	4,47	2,93	2,93	2,93	0,92	0,31	0,15	0,15



Şekil 4.3 Mezun olamadan okuldan ayrılan öğrencilerin okulda bulunma süreleri

4.2. Muğla Üniversitesi Öğrenci Verileri Üzerinde Bilgi Keşfi Bulguları

Model oluşturmak için kullanılacak tabloda bulunan tüm sütunlar, çalışmada kullanılan isimleri ve içerdiği veri türü ile birlikte Ek5'te gösterilmektedir. İlerleyen kesimlerde sütunlara ait bildirimlerde çalışmada kullanılan isimler kullanılacaktır.

4.2.1. Mantıksal eleme

Mantıksal eleme olarak adlandırılan bilgi keşfinin ilk adımında VM modeline katkı sağlamayacağı belirli mantıksal kriterlere göre kesin olan 44 adet sütun başlangıçta elenmiştir. Buna göre;

- Her bir kişiye ait özel değer alan veya modele katkı sağlamayacak kadar gruplanması zor veya hedef sütuna (bağımlı değişkene) etki edecek bir ilgisi olmayan sütunlar çıkarılmıştır. Ek5'te bu sütunların isimlerinin yanına * simgesi getirilmiştir (BABAADI ve EVIL ve VERILYER gibi).
- Her bir kayıt için aynı değerleri içeren benzer sütunlar veya aynı görevi yerine getirebilecek sütunlardan tablodan eleneceklerin isimlerinin yanına ** simgesi, kalacakların isimlerinin yanına ♣ simgesi getirilmiştir (DOGYERI yerine DOGYERIL kalması gibi).
- Tüm tablo boyunca tek değer alan sütunlar, tablodan çıkarılmıştır. Ek5'te bu sütunlar, *** simgesi ile gösterilmiştir (FACNAME gibi).

4.2.2. Veri temizleme işlemleri

Veri hazırlama bölümünün ilk adımı olarak "Boş Değer Yüzdesi" sekmesinde bulunan görev bulunur. Varsayılan olarak %60'ın üzerinde boş değer içeren toplam 42 adet sütun bu sekmede yer alan *FlexGrid* denetiminde görülmüştür. Burada %60 seçimi bir inisiyatiftir. Daha yüksek seçimde daha az, daha düşük seçimde ise daha çok sütun tablodan çıkacaktır. Bu sütunlardan 19'u tamamen boş değer içermekte, 23'ü ise sütun boyunca %60'dan fazla boş değer içermektedir. Tablo 4.6'da modele gelinmeden tablodan çıkarılması gereken bu sütunlar gösterilmektedir.

Tablo 4.6. Yüzde 60'tan fazla boş değer içeren sütunlar

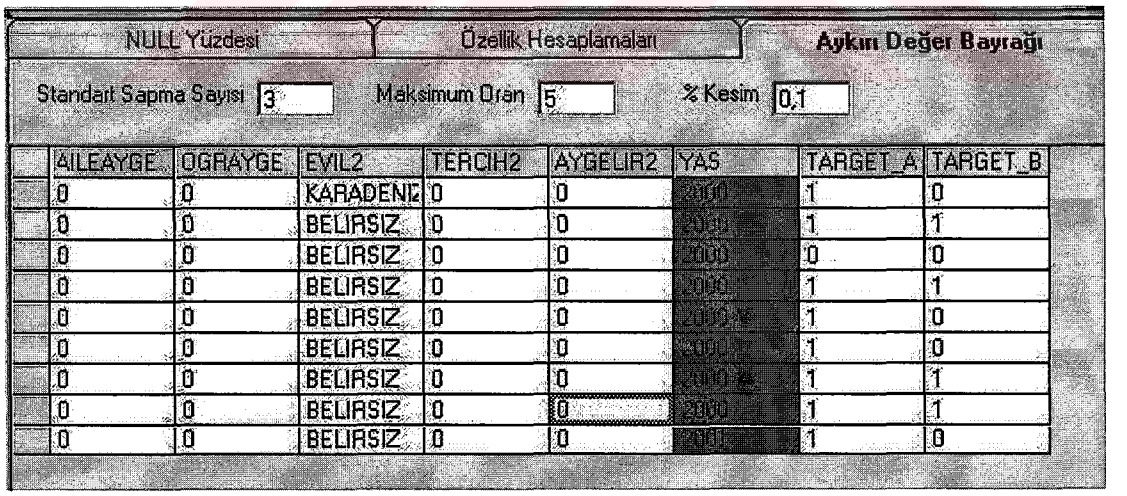
Tamamen boş değer içeren sütunlar (19 adet)		**%60'dan fazla boş değer içeren sütunlar (23 adet)**	
DINI	OLAKAORT	CEZA	OSS_M
OKUMATUR	OLBIYILI	OKULTURU	OSS_S
AILEKISI	GMENSAY	YABDIL	EVDURUMU
OSSDATE	GMENNOT	ONLISMEZ	REGNOTE
OKULAGIR	GMENKIGE	MEZYILI	ARABANOT
HAZIRLIK	ARABASAY	BABAMES	GELIRKAY
MEZDEREC	YAKSODUR	ANNEMES	OGRGEKAY
OYSGRADE		BABASOSD	OGREHLIY
ANNESOSD		BABAYASA	EXPR1
BABABYKS		ANNEYASA	DGSTERCS
CAPACITY		EVULKE	DGSYPUAN
HAZDEREC		OSS_EW	

Veri temizlemenin ikinci adımı olarak gerçekleştirilen sütun özelliklerinde sayısal değer içeren (numeric) sütunlar için beş özellik değeri elde edilmiş olup, Tablo 4.7'de verilmiştir. Buna göre; ANAHTAR tablonun anahtar sütunu olduğu için içindeki farklılık sayısının kayıt sayısı kadar olması normaldir. TERCSIRA ve AILEAYGE sütunlarının gruplanması gerekmektedir. Aksi taktirde modele katkıları olamaz. OGRAYGE de AILEAYGE benzeri bir sütundur; ancak 0 ve 250 gibi sadece iki değer taşıması, gruplanmamış bir sütun için kullanışsızlığı göstermektedir. Diğer sütunlar ise normal gözükmektedir.

Tablo 4.7. Sütun özellikleri

Sütun Adı	Maksimum	Minimum	Ortalama	Standart Sapma	Farklılık Sayısı
ANAHTAR	380654	3364	16270	35952	4403
OKULBIR	2	0	0	0	2
STUDYEAR	12	0	4	2	13
TERCSIRA	26	0	4	6	24
AILEAYGE	900000000	0	48108488	98001008	249
OGRAYGE	250	0	0	4	2
PROGTYPE	2	0	0	0	3
SEMCOUNT	23	0	8	4	23
SEMRECNO	8	1	6	2	6
YEARECNO	12	1	4	2	12

Son temizleme görevi olan aykırı değer işaretlemesi; standart sapma cinsinden uzaklık 3, maksimum değer 5 ve % kesim olarak %0.1 belirlenerek gerçekleştirildiğinde Şekil 4.4'teki durum oluşmuştur. Sonuçta, YAS sütununa ait 9 satırda aykırı değer taşıyan hücre bulunmuştur. Yapılan incelemede hatalı hücre değerleri içeren kayıtların genellikle diğer sütunlarda da boş veya tutarsız değerler içerdiği gözlendiğinden bu kayıtlar tablodan tamamen çıkarılmıştır.



Şekil 4.4 Aykırı değer işaretlenmiş hücreler

4.2.3. Gerçekleştirilen sütun dönüşümleri

Tablo 4.8 dönüşüme uğrayan sütunların dönüşüm öncesi ve sonrası özellikleri göstermektedir. Sayısal değer içeren sütunlardan AILEAYGE, 249 farklı değere sahiptir. Bu değerler 0 ile 900000000 arasındadır. Dönüştürmede bu aralık 5 eşit parçaya bölünerek gruplama gerçekleştirilmiştir. TERCSIRA, 24 farklı değere sahiptir. Yine 5 eşit parçaya bölünerek gruplanmıştır.

Öğrencinin okula başlama yaşı, hazır bir veri olarak bulunmadığından okula kayıt tarihlerini tutan REGDATE sütununun yıl kısmı ile doğum tarihi değerlerini tutan DOGTARIH sütun değerlerinin farkını alan bir işlem yapılarak YAS adlı yeni bir sütun oluşturulmuştur.

ORTALAMA öğrencinin başarısını ölçmede en iyi sütundur. Bu nedenle hedef sütun olarak kullanılmıştır. Ancak dönüşüm öncesi 0.0 ile 4.0 arasında aldığı ondalık değerler ile gruplamaya ihtiyaç duymaktadır. Bu sütun için eşit parçaya bölmek düşüncesi yerine 2.0'ın altını başarısızlığı ifade edecek 0 değeri verilerek bir grup, 2.0 ve üst not değerlerine 1 değeri verilerek diğer bir grup oluşturularak HEDEF_A sütunu elde edilmiştir. ORTALAMA sütunu benzer bir şekilde 3.0'ın altı için 0 ve 3.0 ve üzeri notlar için 1 değeri ile verilerek, üstün başarılıların tespiti için kullanılacak HEDEF_B sütununda tekrar gruplanmıştır. HEDEF_A ve HEDEF_B sütunları için 0 değeri *Negatif*, 1 değeri *Pozitif* değerdir.

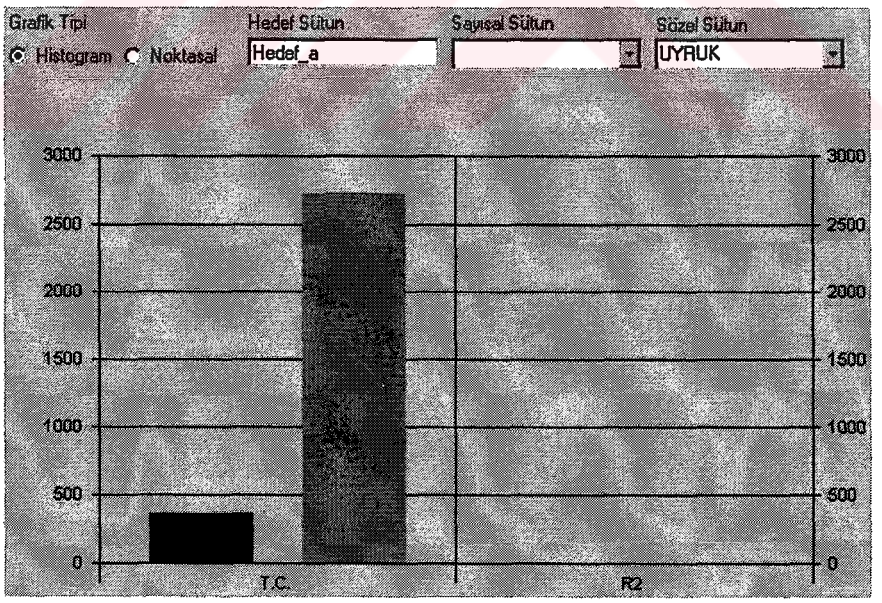
IL ve EVIL benzer değer içeren sütunlar olarak "sadece biri tercih edilebilirdi" diye düşünülebilir; ancak IL, nüfusa kayıtlı olduğu dolayısıyla doğduğu il verilerini içermekte, EVIL ise ailesinin yaşadığı yeri ifade etmektedir ki; bu iki sütun arasında göçlerden kaynaklanan büyük oranda farklılıklar gözlenmektedir. Bu nedenle, bu iki sütunda ülkeyi coğrafi olarak yediye bölen, bölgesel olarak gruplama ile dönüşüme uğratılmıştır.

Tablo 4.8 Dönüşümde kullanılan sütunlar

Dönüşümde kullanılan Sütunlar			**Dönüşüm sonrası oluşan yeni sütunlar**		
Sütun Adı	Veri tipi	Aldığı farklı değer sayısı	Sütun Adı	Veri tipi	Aldığı farklı değer sayısı
IL	varchar	88	IL2	varchar	8
EVIL	varchar	325	EVIL2	varchar	8
TERCSIRA	smallint	24	TERCIH2	smallint	6
AILEAYGE	Float	249	AYGELIR2	float	5
DOGTARİH	datetime	2699	YAS	decimal	31
REGDATE	datetime	135			
ORTALAMA	Float	349	HEDEF_A	decimal	2
			HEDEF_B	decimal	2

4.2.4. Grafiksel incelemeler

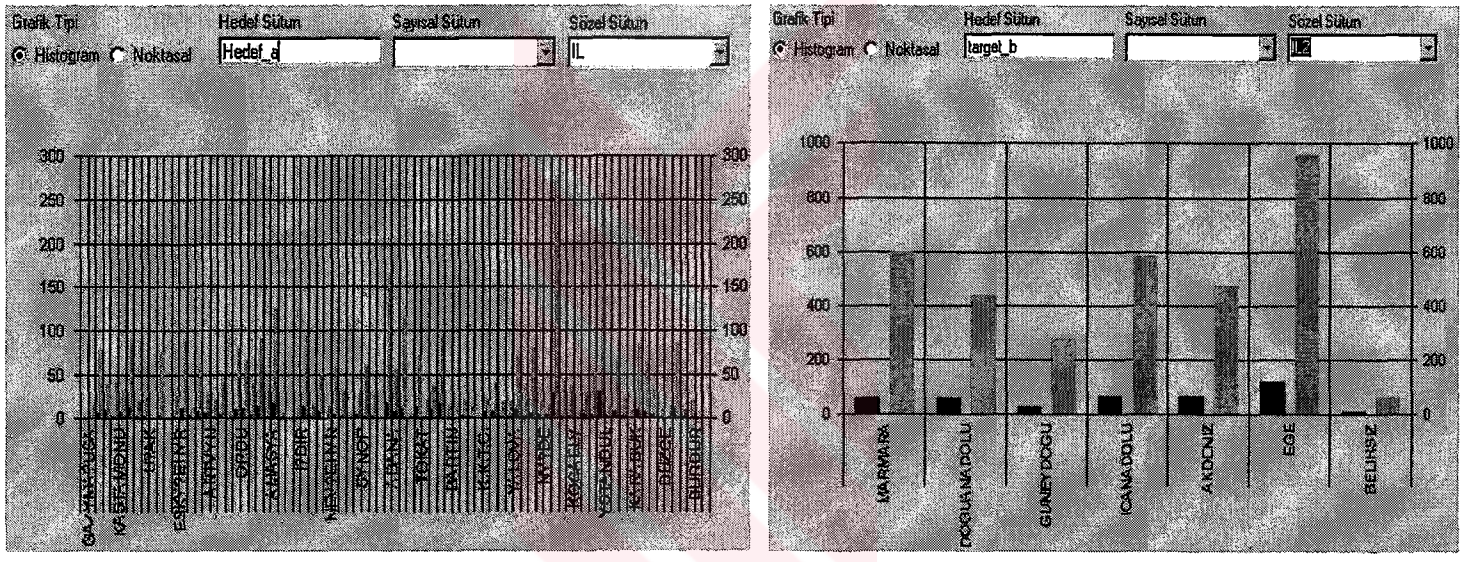
İnceleme aşaması, grafiksel ve sayısal gösterimi kapsamaktadır. Çalışmada hedef sütun olarak belirlenen HEDEF_A ve HEDEF_B'ye diğer karakter tabanlı ve az sayıda farklı değer almış sayısal sütunların nasıl etkilediği histogramlar ile görülebilmektedir.



Şekil 4.5 Uyruk sütunun histogram görüntüsü

Histogram incelemelerinde; REGDATE, DREGDATE ve DOGTARIH; Hedef sütuna etkileri oluşturulan görünmemektedir. Dolayısı ile modele eklenmeyeceği kararına varılmıştır. MEDENHAL, UYRUK, OKULBIR ve DIPNOTU ise tek bir değer üzerine yoğunlaşmanın yaşandığı sütunlardır. Modele ancak çok az bir etki yapabilirler. Bu nedenle modele eklenmemiştir. Örneğin Şekil 4.5'te UYRUK sütununun neredeyse tamamının tek bir değere sahip olduğunu ve modele etki edemeyeceğini ifade etmektedir.

Şekil 4.5 İl değerlerini ülkenin coğrafi bölgelerine gruplama önce ve sonrasındaki hedef sütuna olan etkisini ifade eden histogram görüntüsünü vermektedir. Sütunun aldığı değerlerin hedef sütun boyunca yansız davranmadığı ve tek bir değerin diğer değerlerin etkisini yok edecek derecede baskın olmadığından dolayı gerçekleştirilen dönüşüm sonrası gruplama, anlamlıdır. Etki eden faktörün anlamlı bir şekilde azalması sütun bazında modele olan etkiyi artıracaktır.

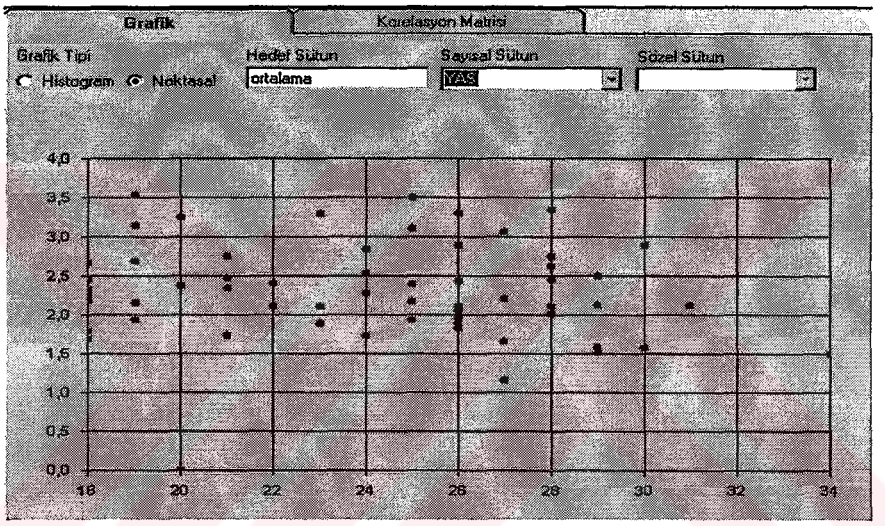


(a) Gruplama öncesi

(b) Gruplama sonrası

Şekil 4.6 İl değerlerini bölgesel bazda gruplama önce ve sonrası histogram görüntüsü

Noktasal grafikler ile yapılan incelemeler, hedef sütunun seçilen sütundaki değerler boyunca, nasıl dağıldığını göstermektedir. Şekil 4.7'de YAS sütununun ORTALAMA sütununa karşı noktasal grafikte dağılımını göstermektedir. Normal olarak 18-23 yaşları olması gereken yoğunluk, bulunmadığı gibi; 18-31 yaşları arasındaki dengeli dağılım, gerçeği yansıtmamaktadır. Buradan REGDATE ve/veya DOGTARIH sütun değerlerinin hatalar içerdiği ve YAS sütununun modele eklenmemesi gerektiği sonucuna varılabilmektedir. Bu kısımda yine OKULBIR ve PROGTYPE sütunlarının da modele pek katkı sağlamayacağı sonuçlarına varılmıştır.



Şekil 4.7 YAS sütununun noktasal grafik görüntüsü

4.2.5. Korelasyon matrisi ve incelemesi

Korelasyon matrisinde, iki ayrı hedef sütun olan HEDEF_A ve HEDEF_B'ye diğer sayısal sütunların tek tek nasıl etkilediği görülebilmektedir. Burada bir inisiyatif alınarak ±0,01 değeri, çalışmada alt sınır alınmış olup, bu değerin altındaki etki ihmal edileceği belirlenmiştir. Anahtar sütun olan ANAHTAR ve hedef sütunların kendileri model oluşturmada kullanılacağı için bu kuralın dışında tutulmuşlardır.

Tablo 4.9 Korelasyon matrisi

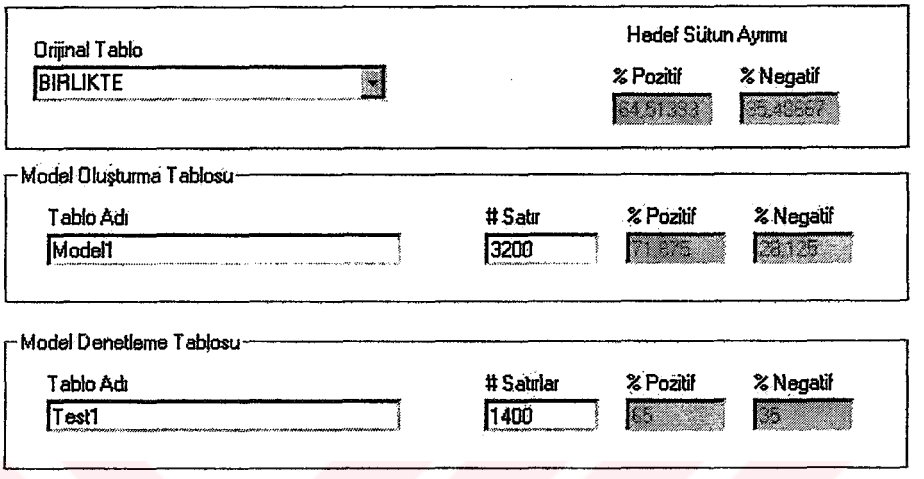
HEDEF SÜTUNLAR	GİRİŞ SÜTUNLARI			
	ANAHTAR	DIPNOTU	OKULBIR	TERCSIRA
HEDEF_A	-0,0268	0,0438	0,0022	0,0462
HEDEF_B	-0,0309	-0,0274	0,0134	0,0466
	TERCIH2	AILEAYGE	AYGELIR2	IDCODE
HEDEF_A	0,0451	-0,0173	0,0066	0,045
HEDEF_B	0,049	0,0468	0,0487	-0,02
	PROGTYPE	SEMCOUNT	SEMRECNO	YEARECNO
HEDEF_A	0,0073	0,0178	-0,0025	0,2069
HEDEF_B	0,0221	-0,0135	0,0184	-0,0073
	HEDEF_A	HEDEF_B		
HEDEF_A	1	0,308		
HEDEF_B	0,308	1		

Bu durumda; OKULBIR, AYGELIR2, SEMRECNO ve PROGTYPE 1.Modele ve YEARECNO ise 2.Modele dahil edilmeyeceği Tablo 4.9'daki korelasyon matrisinden görülebilmektedir.

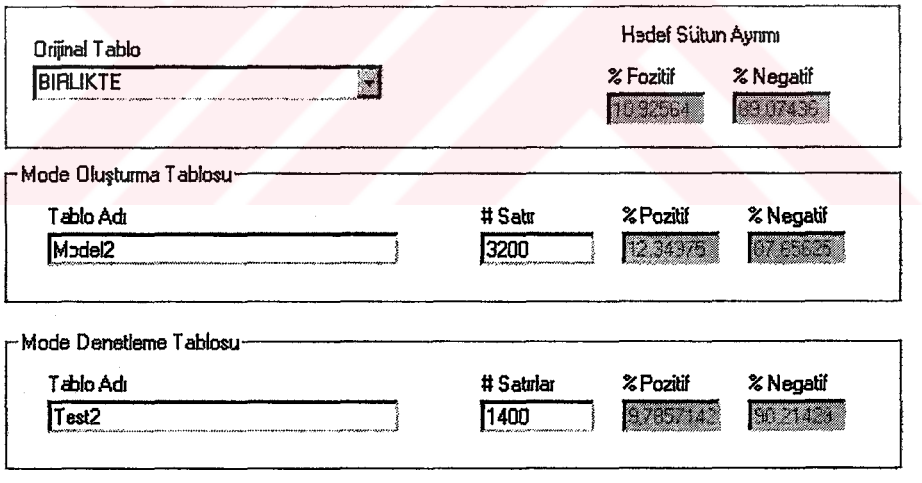
4.2.6. Tablo ayırma işlemi

Kullanılan tabloda modellerde bulunması uygun görülen sütunlar ve bunların modele eklenirken kullandıkları parametreler ile birlikte Tablo 4.8 ve Tablo 4.9'da gösterilmektedir.

Ayırma işleminde yine bir inisiyatif alınarak, model tablosu ile test tablosunun barındıracağı kayıt sayısını belirleyen oran belirlenirken model oluşturmak asıl olduğu için yapılan çalışmada yaklaşık %70 model için, %30 ise test tablosu için kayıt bulundurulmak tercih edilmiştir. Bu da model tablosu için 3200, test tablosu için 1400 kayıt demektir. Oluşturulan model ve test tabloları ve orijinal tabloda hedef sütundaki pozitif ve negatif değerlerin yüzdesi Şekil 4.8 ve Şekil 4.9'da görülebilmektedir. Bu yüzdeler ayrım sırasında model ve test tablolarının orijinal tabloyu ne kadar iyi temsil ettiği ve kayıt seçiminde kullanılan rasgele işlemin kalitesini göstermektedir. Görüldüğü gibi; yeni oluşturulan tablolardaki yüzde değerleri Model1 tablosundaki 7 puanlık fark haricinde orijinal tablodakilere oldukça yakın durumdadır.



Şekil 4.8 1.Model için ayırma işlemi ve hesaplanan yüzdeler



Şekil 4.9 2.Model için ayırma işlemi ve hesaplanan yüzdeler

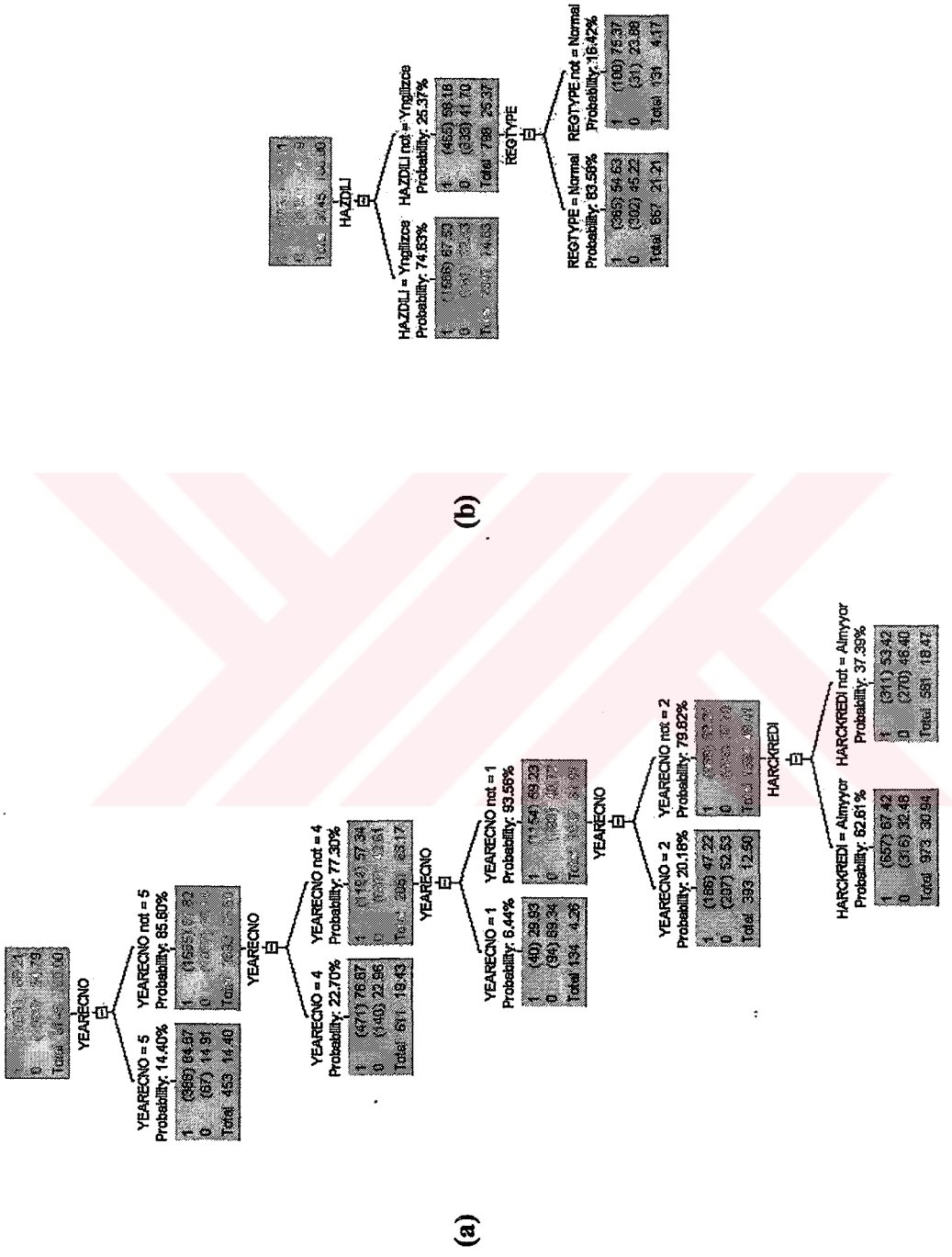
Tablo 4.10 1. ve 2.Modele eklenen sütunlar ve parametreleri♣

NO	SÜTUN ADI	İÇERDİĞİ VERİ	VERİ TİPİ	KULLANIM	İÇERİK TİPİ	ANAHTAR
1	*ANAHTAR*	Anahtar sütun	single	-	-	evet
2	**HEDEF_A***	Not ortalaması 2.0 ve üstü ise 1, değilse 0	single	predictable	Discrete	-
3	**HEDEF_B****	Not ortalaması 3.0 ve üstü ise 1, değilse 0	single	predictable	Discrete	-
4	HAZISTEK	Hazırlık isteği (E/H)	varchar	input	Discrete	-
5	ASKERLIK	Askerlik durumu	varchar	input	Discrete	-
6	CINSIYET	Cinsiyet	varchar	input	Discrete	-
7	HARCKRED	Harç kredisi (alıyor/almıyor)	varchar	input	Discrete	-
8	EDUCTYPE	Eğitim türü (Normal/II.Öğr.)	varchar	input	Discrete	-
9	DEPTNAME	Bölüm adı	varchar	input	Discrete	-
10	IDCODE	Bölüm ve Eğitim Türü beraber	varchar	input	Discrete	-
11	REGTYPE	Okula kayıt olma türü	varchar	input	Discrete	-
12	OKULKOLU	Lise okul türü	varchar	input	Discrete	-
13	HAZDILI	Hazırlık dili	varchar	input	Discrete	-
14	PROGTYPE**	Program türü	İnteger	input	Discrete	-
15	OKULBIR**	Okul birinciliği	integer	input	Discrete	-
16	SEMRECNO**	Okulda bulunduğu yarıyıl	integer	input	Discrete	-
17	SEMCOUNT	Yarıyıl toplamı	integer	input	Discrete	-
18	YEARECNO*	Okulda bulunduğu yıl	integer	input	Discrete	-
19	AILEAYGE	Ailenin aylık geliri	single	input	Discrete	-
20	IL2	Doğduğu ilin dahil old. bölge	varchar	input	Discrete	-
21	EVIL2	Yaşadığı ilin dahil old. bölge	varchar	input	Discrete	-
22	TERCSIRA	Tercih sırası	integer	input	Discrete	-
23	TERCIH2	Tercih sırası	integer	input	Discrete	-
24	AYGELIR2**	Aile aylık geliri	single	input	Discrete	-

♣Sadece 1.Modele eklenenlerin sütunların isimlerinin yanına *, sadece 2.Modele eklenenlerin sütunların isimlerinin yanına ** simgesi getirilmiştir.

4.2.7. Model oluşturulurken kullanılan sütunlar ve elde edilen model görüntüleri

Model ve test tabloları oluşturulduktan sonra model tablosu ile model oluşturma işlemine gidilmiştir. Tablo 4.10'da verilen tablo son modellere eklenen sütunları göstermektedir. Görüldüğü gibi, modellere eklenen sütunlar arasında hedef sütunların birbirine benzemesinden ötürü çok az fark bulunmaktadır. Hedef sütunlar haricinde sadece korelasyon matrisinde edinilen bilgiye dayalı olarak, HAZSINAV ve SEMRECNO 1.Modele; YEARECNO ise 2.Modele eklenmemiştir. Bu tablolarda yine sütunların modele hangi parametreler ile eklendiği gösterilmektedir.



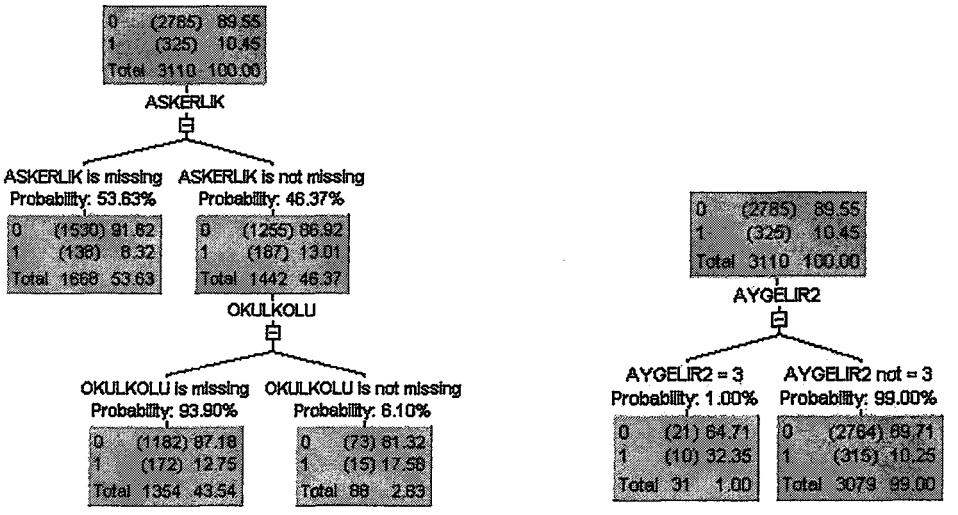
Şekil 4.10 1.Modele ait karar ağacı görünümleri

Buna göre tüm popülasyon içerisinde hedef sütuna etki eden sütunlar önem sırasına göre ilk karar ağacında YEARECNO ve HARCKRED olarak Şekil 4.10a'da görünmektedir. YEARECNO sütununun sırasıyla 5,4,1 ve 2 değerlerinde ayrımlar gerçekleşmiştir. Örneğin kökten ilk ayrım, YEARECNO'nun 5 değerinde gerçekleşmiştir. Bu sütun %14.40 oranla 5 değerini içerir. Bu kısmın %84.87'sini oluşturan 386 kayıtta hedef sütun pozitif değer olan 1; %14.91'ini oluşturan 67 kayıtta ise hedef sütun 0 değerini almıştır. YEARECNO, 5'ten farklı değer alan kayıtlar %85.60 oranla 2692 kişidir. Bunların da %61.82'inde hedef sütun 1; %38.14'ünde hedef sütun 0 değeri almaktadır. YEARECNO'nun 5 değerini ilk ayrım durumuna getiren burada 5 değerinde %84.87 oranında hedef sütunda 1 değerinin alındığının bulunmasıdır ki bu kökte yani tüm tabloda %65.21'dir. Diğer ayrımlar da bu şekilde açıklanabilir.

YEARECNO sütununun başarıya en çok etkili bulunması, mezuniyetin gereği ile hedef sütunun 1 değerinin örtüşmesi sonucudur. Bu nedenle bu iki sütun birbirleri ile bağımlı olmamalarına rağmen bu bir derece beklenen durum olarak görülebilir. Bir sonraki önemli nitelik olarak HARCKRED çok az da olsa hedefe etki etmiş ve pozitif değerde 2 puanlık bir yükseltme gerçekleştirerek ağaca dahil olmuştur.

YEARECNO'nun etkisinin ihmal edildiği yani modelden çıkarıldığı durumda ise, Şekil 4.10b'deki durum gözlenmiştir. Burada HAZDILI ve REGTYPE hedef sütunun tahmin etmede en etkili sütunlardır. HAZDILI (hazırlık dili) *İngilizce* olması ve REGTYPE (okula kayıt tipi) yatay ve dikey geçiş gibi *Normal* değerden farklı olması öğrenci başarısında etkili olduğunu karar ağacında görülmektedir.

2.Modele ait iki adet karar ağacı görünümleri Şekil 4.11a ve Şekil 4.11b'de verilmiştir. İlk model görünümü olan Şekil 4.11a, VBK'nın insan-bilgisayar birlikteliği ile gerçekleştirilmesi gerekliliğini göstermesi açısından anlamlıdır. Burada ASKERLIK ve OKULKOLU sütunlarında ayrım gözlenmektedir. Ancak ayrımı oluşturan etki boş değerlerdir. Her ne kadar veri hazırlama bölümünde bunun önüne geçmek için çeşitli filtreleyici işlemler gerçekleştirilmesine rağmen boş değer içeren her bir sütunun silinmemesi durumunda her zaman için karşılaşılabilecek bir durumdur.

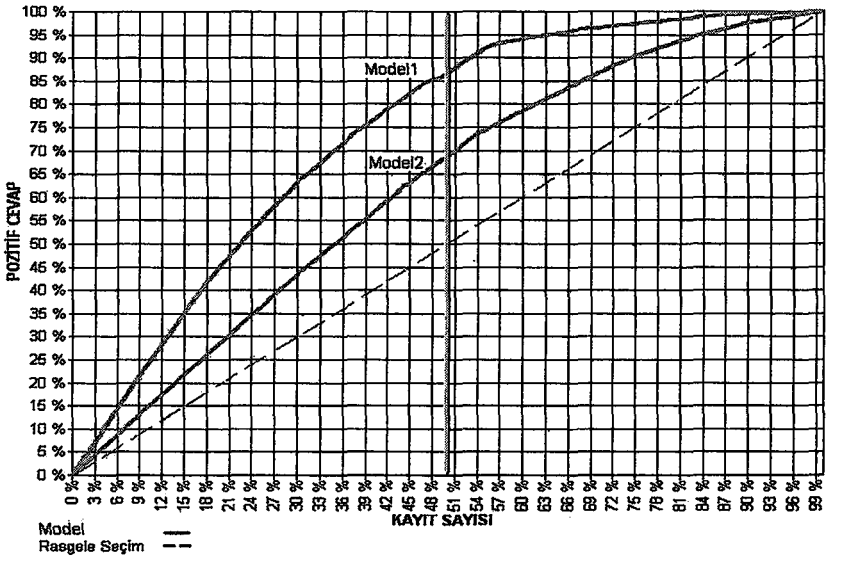


Şekil 4.11 2.Modele ait karar ağacı görünümleri

Modelden bu anlamsız etki eden sütunlar, geri çekildiğinde oluşan model görünümü ise Şekil 4.11b'de verilmiştir. Tüm popülasyon üzerinde AYGELIR2 verideki ilk ayrım nerede olacağına karar vermekte en önemli etken olarak karar ağacı algoritması tarafından seçilmiştir.

AYGELIR2 sütunu öğrencinin ailesinin aylık gelirinin tutulduğu AILEAYGE sütunundan dönüştürülerek oluşturulmuştur. AYGELIR2 sütununun 3 değeri, 400 ile 600 milyon arasını göstermektedir. Bu sütundaki maksimum değer, 900 milyon olduğuna göre; yüksek derecede bir başarıya, orta ve ortanın üstü bir gelire sahip kitlenin daha yatkın olduğu yargısına varılabilir.

Modelde gözükmeyen sütunların hedef sütuna bir etkisi yoktur denilemez. Modele eklenen tüm sütunlar hedef ile ilgili düşük de olsa bir etkiye sahip olduğu veri hazırlama bölümündeki aşamalarda gözlenmiştir. Örneğin korelasyon matrisinde hedef sütuna etkisi çok az gözlenen bazı sütunlar eklenmemiştir.



Şekil 4.11 Oluşturulan modellere ait kaldıraç grafikleri

Şekil 4.10'da oluşturulan modellere ait kaldıraç grafikleri verilmiştir. Tüm kayıtlar içerisinde %50 sayıda kayıt rasgele seçildiğinde tüm pozitif cevapların %50'sine ulaşılabilmektedir. Bu karakteristik bir eğridir; çünkü modelde kullanılan tablodaki kayıtlardan gelen toplam pozitif cevapların sayısı her zaman aynıdır. Eğitilen modeller kullanıldığında bu değer, 1.Model için %87'ye; 2.Model için %68'e ulaşmaktadır. Dolayısıyla 1.Model için kaldıraç oranı 87/50=**1,74**; 2.Model için 68/50=**1,36** olmaktadır. Bu durum, modelin seçimde aktif rol aldığını göstermektedir.

Çalışmada oluşturulan modeller aynı tablo üzerinden gerçekleştirildiğinden grafiklerde büyük benzerlikler gözükmektedir. Ancak hedef sütunların ORTALAMA sütunundan dönüşümünde kullanılan kuralların farklı olması ve bir modelde bu hedef sütunlardan sadece birinin bulunması nedeni ile birbirlerinden bağımsız olarak düşünülmektedir. 2.Modelin tahmin etme becerisinin düşük oluşu ise hedef sütunda yer alan pozitif değerlerin sayıca 1.Modelden daha az oluşundandır. 1.Model için pozitif değer 2.0'a eşit veya daha yüksek not sahibi olmak iken; 2.Model için pozitif değer, 3.0 ve üzeri not değerine sahip olmaktır. 1.Model, %65'lik; 2.Model, %11'lik bir pozitif değer oranına sahiptir. Dolayısıyla pozitif oranın azlığı modelin eğitiminde oluşturulan ilişkilerin kurulmasını zorlaştırıcı bir etki etmektedir.

5. SONUÇLAR ve TARTIŞMA

Bu çalışma, Muğla Üniversitesi öğrenci verileri kullanılarak gerçekleştirilen örnek bir bilgi keşfi uygulamasıdır. Burada, yüksek öğrenimdeki öğrenci performansını değerlendirmede veri madenciliği tekniğini kullanan veri yönelimli bir yaklaşım sunulmaktadır. Gerçekleştirilen bu çalışmanın, VM sürecinin veritabanı yönetim sistemlerine entegre edildiği gelecekteki bazı çalışmalara yön göstereceği düşünülmektedir.

Çalışmanın ana öğelerinden olan geliştirilen program, MÜKÜP, VBK sürecini en temel düzeyde gerçekleştirebilmektedir. *Veri hazırlama* ve *Model oluşturma* bölümleri kullanılan verilerin ve üzerinde çalışılan problemin ihtiyaçları düşünülerek oluşturulmuştur. Bu nedenle farklı veriler ve problemlere karşı yenilemelere ve eklentilere ihtiyaç duymaktadır. Ve yine çözüm geliştirme aşamasında program, bir çok kez SQL sunucu ve Analiz Hizmetlerinin işlevselliklerini kullanmakta ve bu işlevselliklere Transact-SQL veya VBScript gibi dillerde yazılmış program parçaları ile ulaşmaktadır. Bu nedenle kullanıcıların bu teknolojilere yatkın olması gerekmektedir.

Çalışmanın başlangıcında, Muğla Üniversitesi'ndeki öğrencilerin kendi geçmiş verilerinden yola çıkılarak, veritabanı sorguları gerçekleştirilmiştir. Burada bazı önemli demografik yüzdeler, bölüm tercih sıralaması, kayıp ve yerleşim zamanları gibi tanımlama amaçlı çeşitli analizler ile şu andaki durum hakkında daha iyi fikir edinmek hedeflenmiştir. Daha sonra yine aynı veriler kullanılarak öğrencilerin eğitimdeki başarılarını ifade edebilen dönem sonu ortalaması, hedef gösterilerek keşif işlemleri gerçekleştirilmiştir. Burada öğrencilerin eğitimdeki başarı/başarısızlıklarına etki edebilen kişisel özellikleri keşfetmek amaçlanmıştır. Keşif işleminde bir VM tekniği olan karar ağacı algoritması ile sınıflandırma modelleri gerçekleştirilmiştir.

Bu çalışmada oluşturulan birbirinden bağımsız iki sınıflandırma modelinden ilki, 2.0 not ortalamasını; ikincisi ise 3.0 not ortalamasını sınır olarak almıştır. İlk modelde öğrencilerin üniversiteye kayıt olma türü, ikinci modelde ise ailenin aylık geliri özellik değerleri hedefi etkileyen etkenler olarak bulunmuştur. Oluşturulan modellerin denetiminde, kaldıraç grafiği kullanılmış ve modellerin belirli düzeyde tahmin etme yeteneklerinin olduğu tespit edilmiştir.

Çalışmanın bilgi keşfi süreci sırasında, çok sayıda sütun boş ve tutarsız veriler içerdiği için modele eklenememiştir. Bu durum, bu tür bir çalışma için oluşturulan modelin sağlığını doğrudan etkilemektedir. Yine bazı sütunlar gruplanabilmesi çok zor olacak sayıda farklı değerler içermesi de modelde faydalanılmasına engel olmuştur. Bu nedenle; sistemdeki veri ve veride bulunan ilgili niteliklerin sayısında düzenli bir şekilde artırım sağlanılabilirse, üniversite öğrencisinin başarısı hakkında daha iyi tahminler yapılabilir. Öğrencilerin başarısını ölçmede faydalı olabilecek verilerin önemli bir kısmı öğrencilerin üniversiteye kayıt olduğu esnada sahip olunduğundan, bu kısıtlı dönemde öğrencilere ait kişisel bilgileri daha hızlı ve norm olarak toplamaya izin veren optik kodlamanın tercih edilmesi yerinde olacaktır.

Gerçek dünyada öğrencinin performansını birçok faktör etkilemektedir. Bu nedenle, yalnızca iç veriler ile yetinmek yerine, ilgili dış veriler de araştırmaya dahil edilmelidir. Bu veriler var olan sistem ile bütünleştirilebilirse ve öğrenci kendi durumu hakkında zamanında bilgilendirilebilirse, onu başarıya götürebilecek tercihlere yönlendirilebilir veya gerekli görülürse sistem değişikliğine gidilebilir.

Ham verinin elde edilişinden, işlenişine; model oluşturmadan, denetlenmesine kadar VBK sürecinin her safhasında zamanla çeşitli düzenlemelere gidilebilir. Örneğin çalışmanın *veri temizleme* ve *veri ayırma* bölümlerinde bilgi keşfinin ruhunda yer alan ve literatürde kullanıcının seçimine bırakılan değer aralığında bazı inisiyatifler alınmıştır. Eldeki veri ve hedeflenen sütunun durumu gözetilerek belirlenen bu değerler, farklı senaryolar kurularak tekrarlanırsa bazı farklılıklar ortaya çıkarılabilir.

Bu çalışmada yer almayan, öğrencilerin aldıkları zorunlu/seçmeli derslerdeki başarıları incelenirse, öğrencinin almak istediği dersler ile ilgili taşıdığı riskler, problemler ve sonuçlar hakkında daha fazla bilgi verebilen bir danışmanlık gerçekleştirilebilir. Ancak burada eşleştirme kuralları gibi algoritmalar ile çalışma zenginleştirilmelidir. Daha ileri bir adım olarak, danışmanın rolünü alabilen bir uzman sistem geliştirilebilir.

KAYNAKLAR

Adriaans, P., Zantinge, D., *Data Mining*, 1996. Addison Wesley Longman, Edinburg Gate, 159p.

Agrawal, R., Imielinski, T., Swami, A., Mining Association Rules between Sets of Items in Large Databases, *Proceedings of the* 1993 *ACM SIGMOD Conference,* May 26-28 1993, Washington DC, USA, 207-216 pp.

Agrawal, R., Mehta, M., Shafer, J., Srikant, R., Arning, A., Bollinger. T., The Quest Data Mining System Proceedings of 1996, *International Conference on Data Mining and Knowledge Discovery (KDD'96)*, Portland, Oregon,1996

Akpınar, H., 2000. Veritabanlarında Bilgi Keşfi ve Veri Madenciliği, *İ.Ü. İşletme Fakültesi Dergisi*, 29 (1) : 1-22

Almuallim, H., Dietterich, T.G., 1991. Learning With Many Irrelevant Features, *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91)*

Apaydın, E., Z., Veri Madenciliği: Ham Veriden Altın Bilgiye Ulaşma Yöntemleri, Bilişim 2000 Eğitim Semineri, http://www.cmpe.boun.edu.tr/~ethem, 2004 ;

Berry, M. J. A., Linoff, G. 2000. *Mastering Data Mining: The Art and Science of Customer Relationship Management,* John Wiley and Sons, New York, 512 p.

Berson, A., Smith, S. J., 1997. *Data Warehousing, Data Mining and OLAP,* McGraw-Hill, New York, 612p.

Bontempo, C.J., Saracco, C.M., 1995. *Database Management,* Prentice Hall, New Jersey, 390p.

Borgelt C., 2004. APRIORI - Find Association Rules/Hyperedges with Apriori Algorithm, http://fuzzy.cs.uni-magdeburg.de/~borgelt,

Borgelt C., 2004. Decision Tree Induction, http://fuzzy.cs.uni-magdeburg.de/~borgelt,

Brachman, R.J., Khabaza, T., Kloesgen, W., Piatetsky-Shapiro, G., Simoudis, E., 1996. Mining Business Databases, *Communications of the ACM,* 39, (11) : 42-48.

Bulun, M., Tuğ, E., Şakiroğlu, A.M., Tıbbi Veri Tabanlarında Gizli Bilgilerin Keşfedilmesi

Chan, K.C., Wong, A.K.C., 1991. A statistical test for extracting classificatory knowledge form databases. Knowledge Discovery in Databases, Ed., *The AAAI Press,* 107-123 pp.

Chen, M., Han, *J.,* Yu, P.S., 1996. Data Mining: An Overview from Database Perspective, *IEEE Transactions* on *Knowledge and Data Engineering,* 8, (6) : 866-883

De Jong, K. A., 1999. Evolutionary Computation for Discovery, *Communications of the ACM,* 42, (11) : 51-53

Edelstein, 1999. Competitive Intelligence Review, *John Wiley & Sons,* 12, (1) : 32 - 40

Fayyad, U.M., Piatetsky-Shapiro G., Smyth P., Uthurusamy R., 1996. *Advances in Knowledge Discovery and Data Mining,* The MIT Press, Massachusetts, 625p.

Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P. 1996. The KDD Process for Extracting Useful Knowledge from Volumes of Data, *Communications of the ACM,* 39, (11) :27-34

Fayyad, U.M., Uthurusamy R., 1996. Data Mining and Knowledge Discovery in Databases, *Communications of the ACM,* 39, (11) : 24-26

Frawley, W. J., Piatetsky-Shapiro, G., Matheus, J. 1991. Knowledge Discovery in Databses: An Overview. in Knowledge Discovery in Databases, *AAAI Press*, Menlo Park. 1-27.

Fu, L., 1999. Knowledge Discovery Based on Neural Networks, *Communications of the ACM,* 42, (11) : 47-50

Ge, A., 1998. Data Mining Overview and Products", http://scanner-group.mit.edulhtdocs/ thesis/angelagethesis.html

Glymour, C., Madigan, D., Pregibon, D., Smyth, P., 1996. Statistical Inference and Data Mining, *Communications of the ACM,* 39, (11) : 35-41

Goebal, M., Grunwald, L., 1999. A Survey of Data Mining and Knowledge Discovery Software Tools,ACM SIGKDD, 1, (1) : 20-33

Güvenç, E., *Student Performance Assesment in Higher Education Using Data Mining,* Msc Thesis, Bogazici University, 2001.

Hermiz, K.B., 1999. Critical Success Factors for Data Mining Projects, DM *Review,* http://www.dmreview.com

Information Discovery Inc., 1997. A Characterization of Data Mining Technologies and Processes, http://www.datamining.comldm-technology.htm

Kira, K., Rendell, L. 1992. The feature selection problem: Tradational methods and a new algorithm, *In Proocedings of AAAI-92, AAAI Press*,129-134 pp.

Michalski, R.S., Stepp, R., 1983. Automated construction of classifications: Conceptual clustering versus numerical taxonomy, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 5, 396-410 pp.

Mitchell, T.M., *1999.* Machine Learning and Data Mining, *Communications of the ACM,* 42, (11) : 31-36

Munakata, T., 1999. Knowledge Discovery, *Communications of the ACM,* 42, (11) : 27-29

Oğuz, B., Eşleştirme Haznelemesinin Biçimsel Kavram Analizi ile Modellenmesi, Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, 2000 yükseklisans tezi

Özmen, Ş., Veri Madenciliği ve Uygulama Alanları., http://suleozmen.marmara.edu.tr/, 2003

Paul, S., Gautam, N., Balint, R., 2004. Preparing and Mining Data with Microsoft SQL Server 2000 and Analysis Services, 154p.

Quinlan, J.R., 1986. Induction of Decision Trees. *Machine Learning*, vol. 1, pp 81-106

Rigdon, E.E., Bacon, L., 1998. Data Warehousing and Data Mining: Possibilities, Pitfalls, and Implications for Marketing Management and Research, http://www.marketing.gsu.edulmk8700eer/dmgoals7.html,

Sholom, M. W., Indurkhya, N., 1998. *Predictive Data Mining A Practical Guide*, Morgan Kaufmann Pub., San Francisco, 228p.

Soni, S., Tang, Z., yang, J., 2002. Performance Study of Microsoft Data Mining Algorithms, http://research.microsoft.com

Swift, R.S., 2000. Accelerating Customer Relationships:Using CRM and Relationship Technologies, *Prentice Hall PTR*, 512p.

Two Crows Corporation, 1999. Introduction to Data Mining and Knowledge Discovery, *Data Mining* '99: *Technology Report,* http://www.twocrows.com/dm99tr.htm,

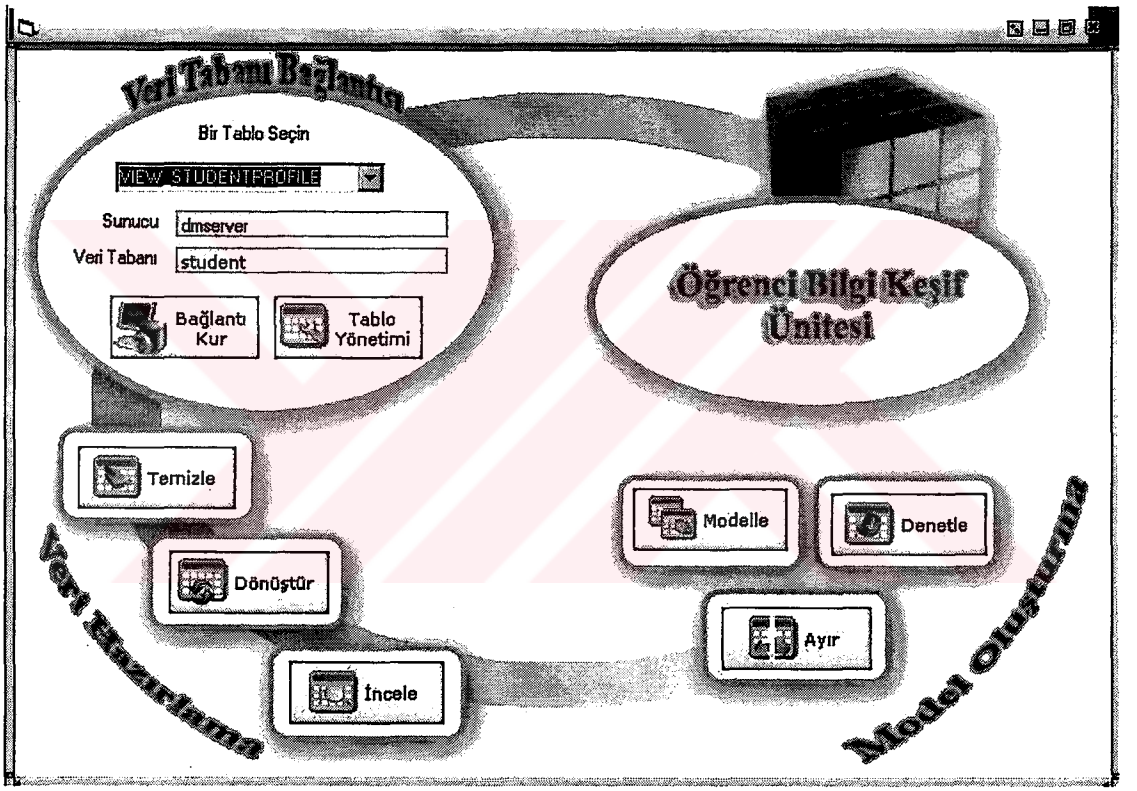
Vahaplar, A., İnceoğlu, M. M., Veri Madenciliği ve Elektronik Ticaret, http://www .inet-tr.org.tr/inetconf7/eposter/inceoglu.doc, 2002

Weiss, S.M., Kulikowski, C.A., 1991. Computer Systems that Learn, *Morgan Kaufmann Pub.*, San Mateo, California.

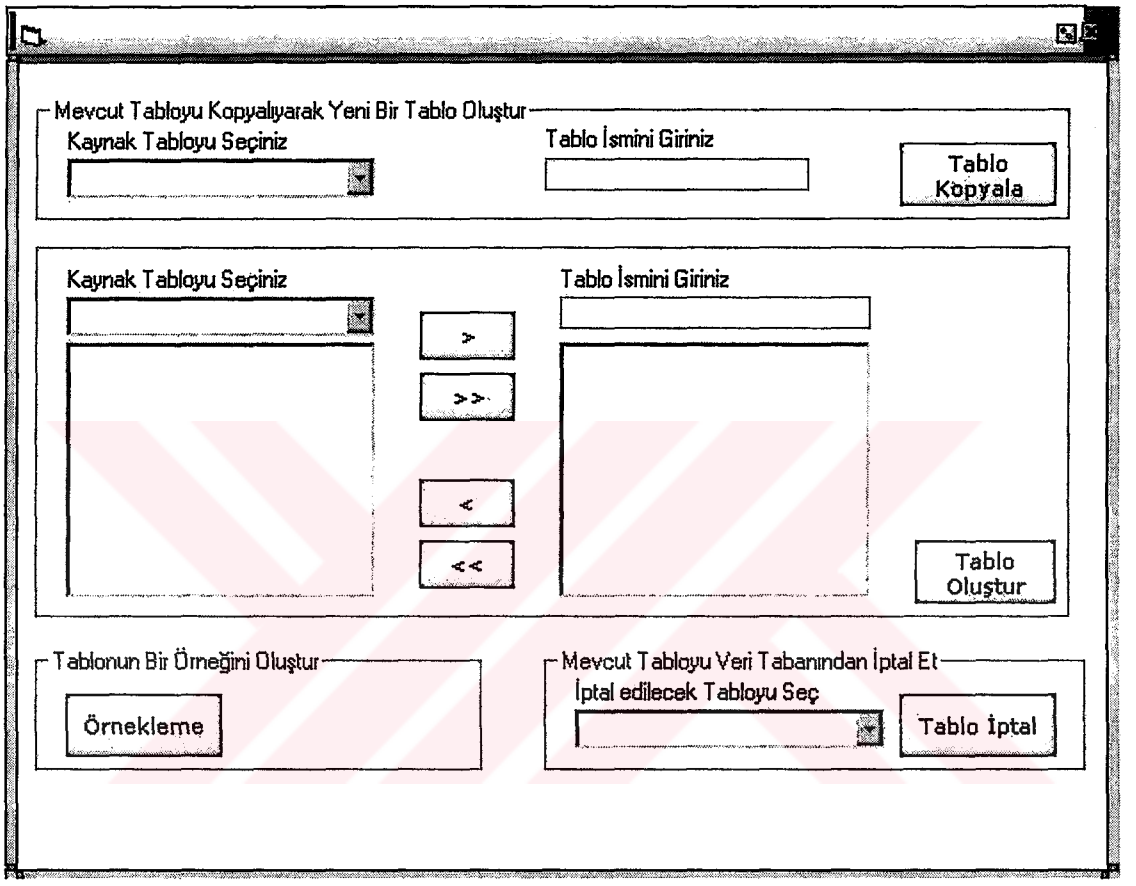
Yurtsever U., *Veri Madenciliği ve Uygulaması*, Yüksek Lisans Tezi, Sakarya Üniversitesi Sosyal Bilimler Enstitüsü, 2002.

EKLER

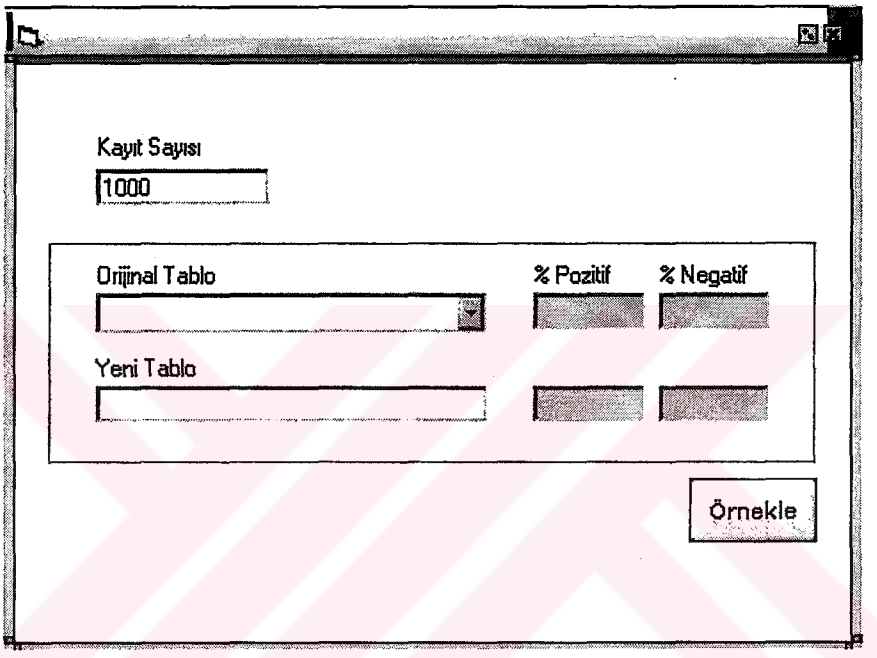
Ek 1. MÜKÜP Form Görünümleri



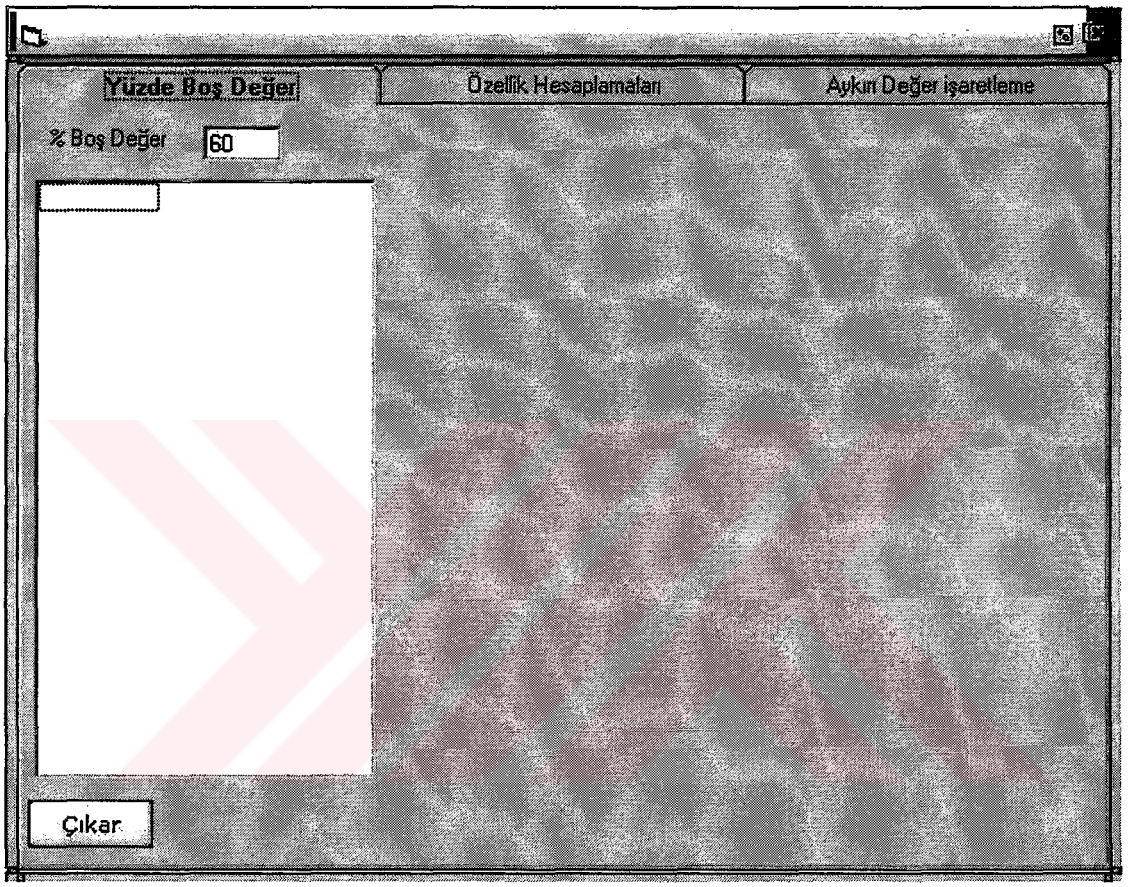
Form 1. MÜKÜP kullanıcı ara yüzü formu



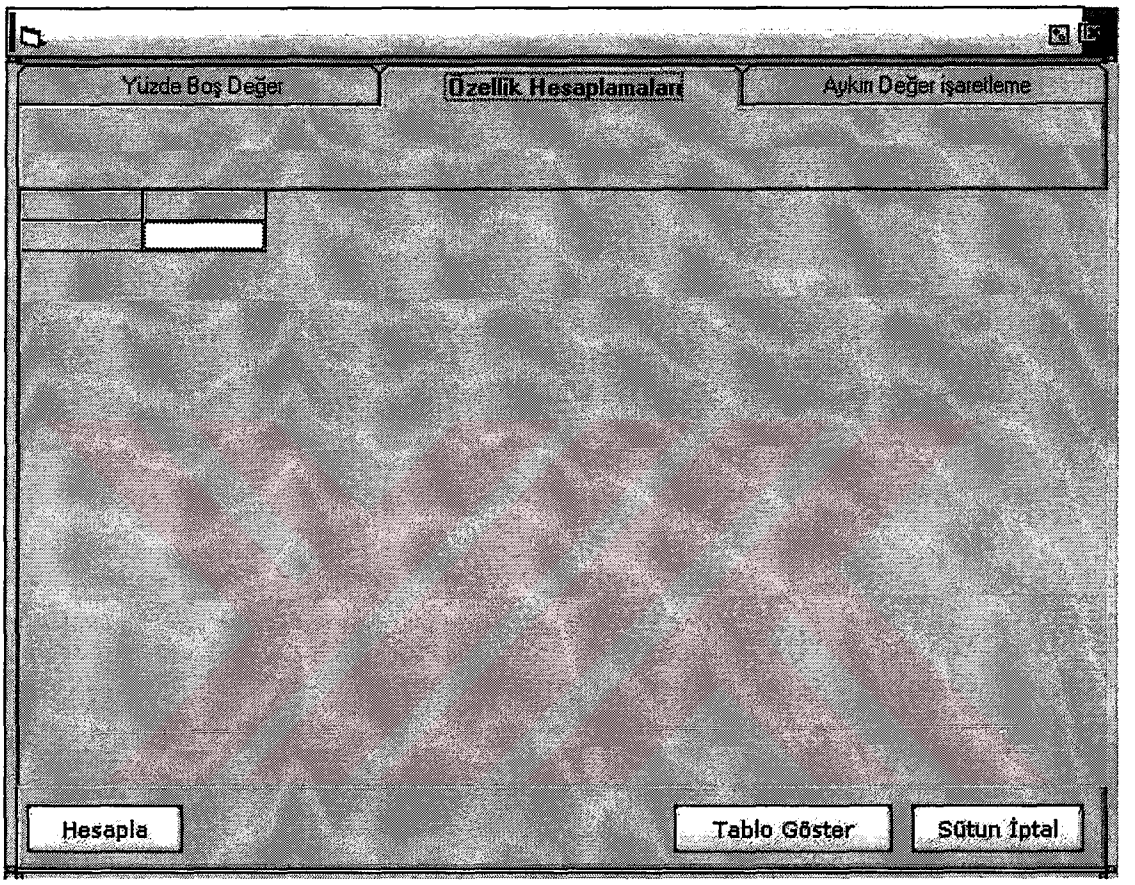
Form 2a. Tablo yönetimi formu



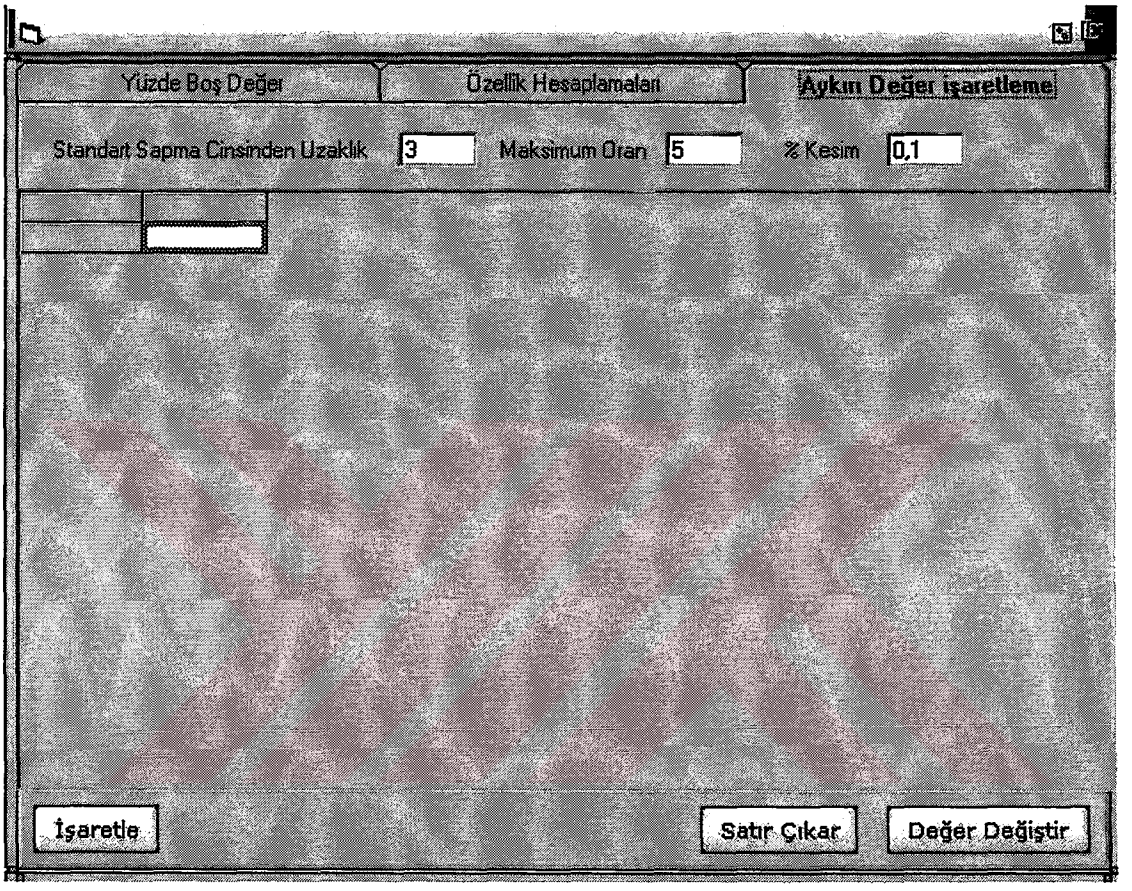
Form 2b. Tablo örnekleme formu



Form 3a. Temizleme formu, Boş Değer Yüzdesi sekmesi

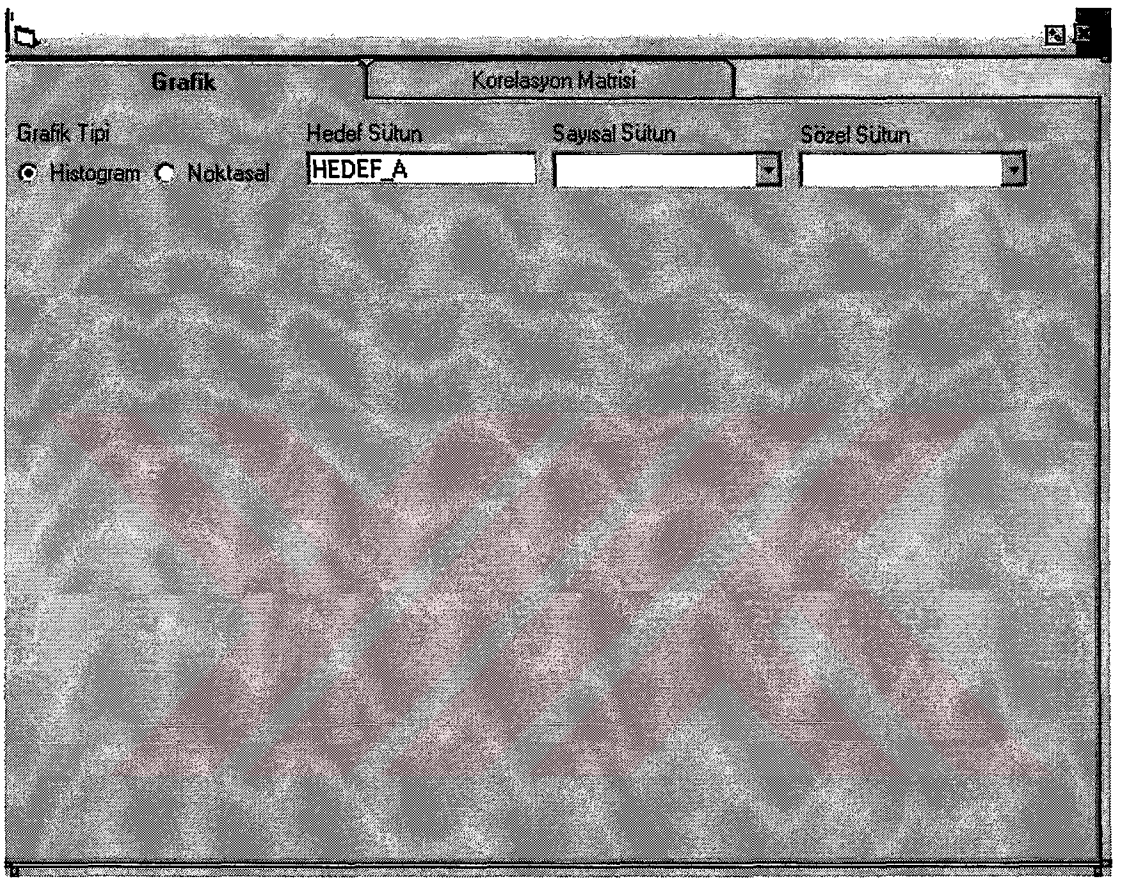


Form 3b. Temizleme formu, özellik hesaplamaları sekmesi

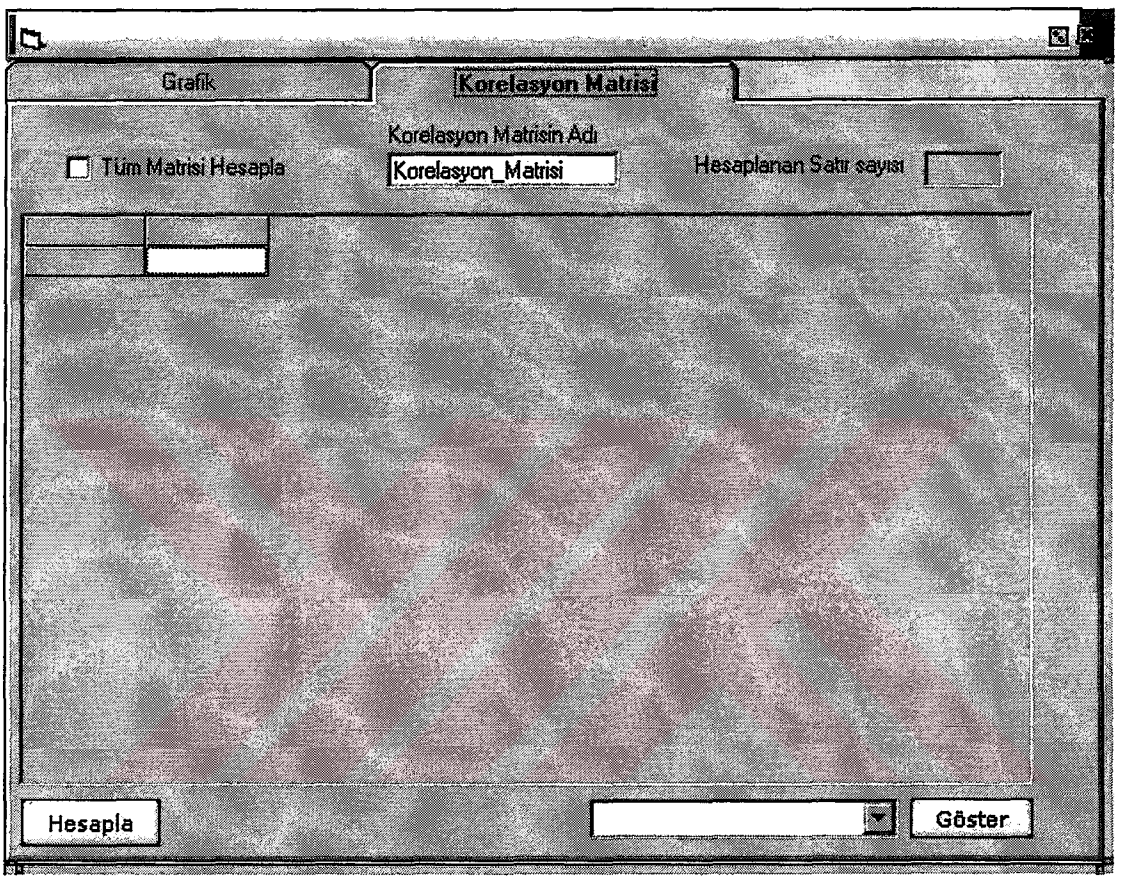


Form 3c. Temizleme formu, aykırı değer işaretleme sekmesi

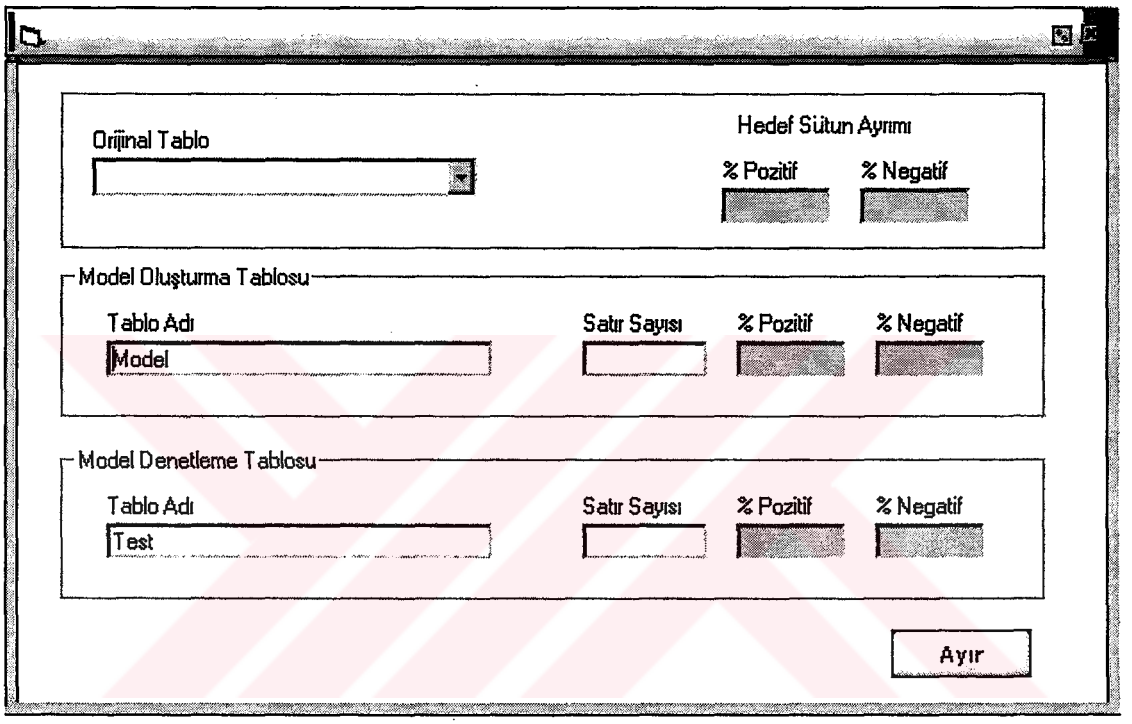
Form 4. DTS Import/Export sihirbazı açılış formu



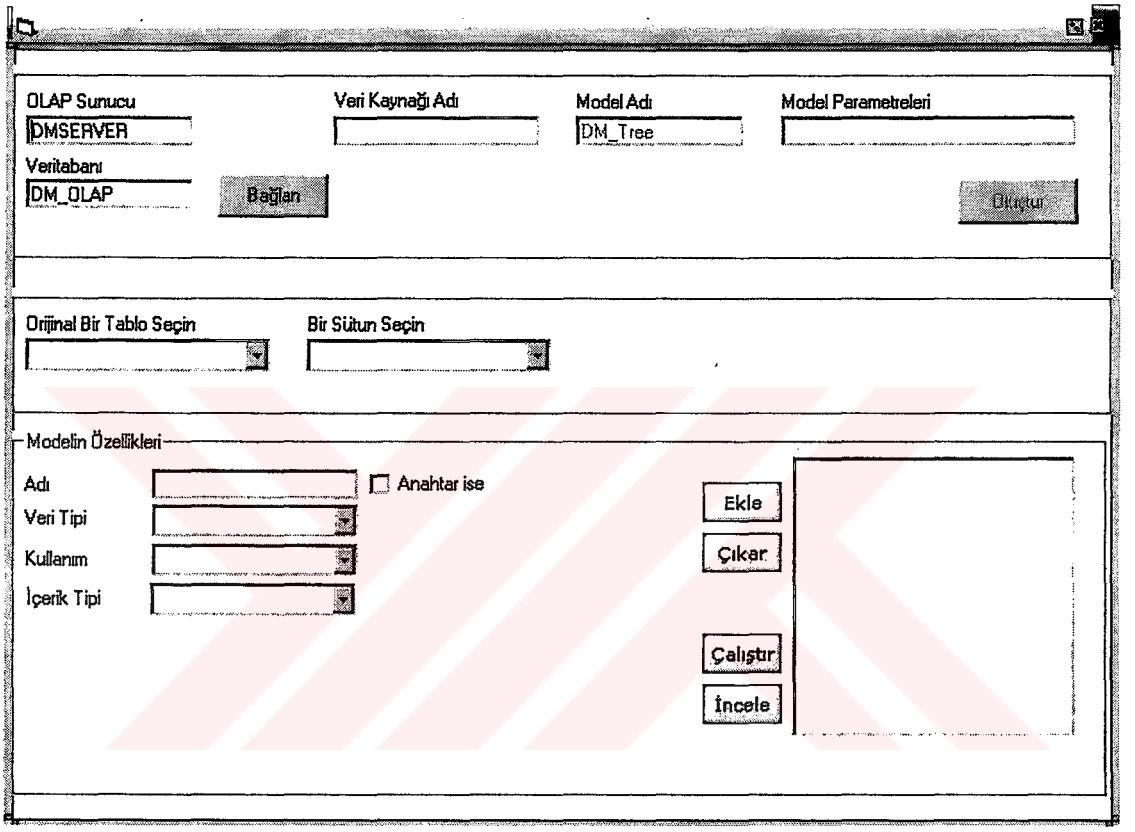
Form 5a. İnceleme formu, grafik sekmesi



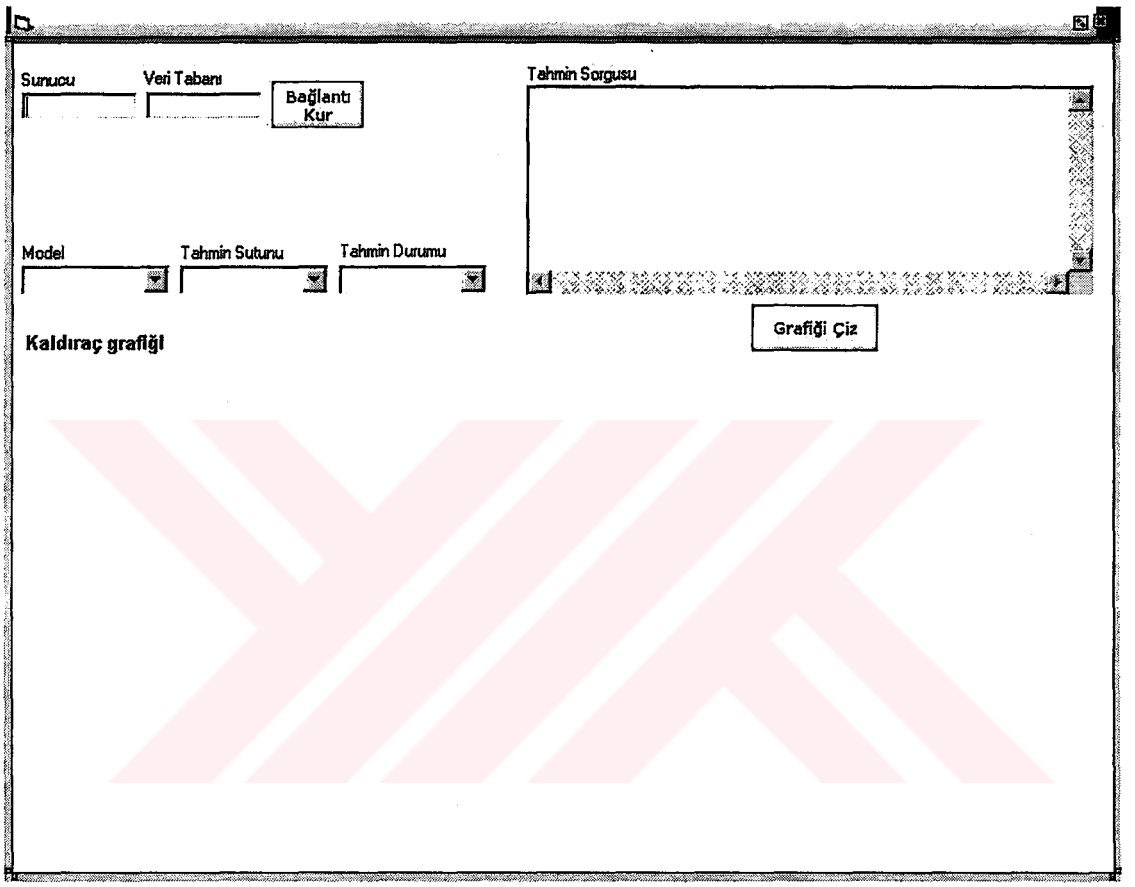
Form 5b. İnceleme formu, korelasyon matrisi sekmesi



Form 6. Tablo ayırma formu

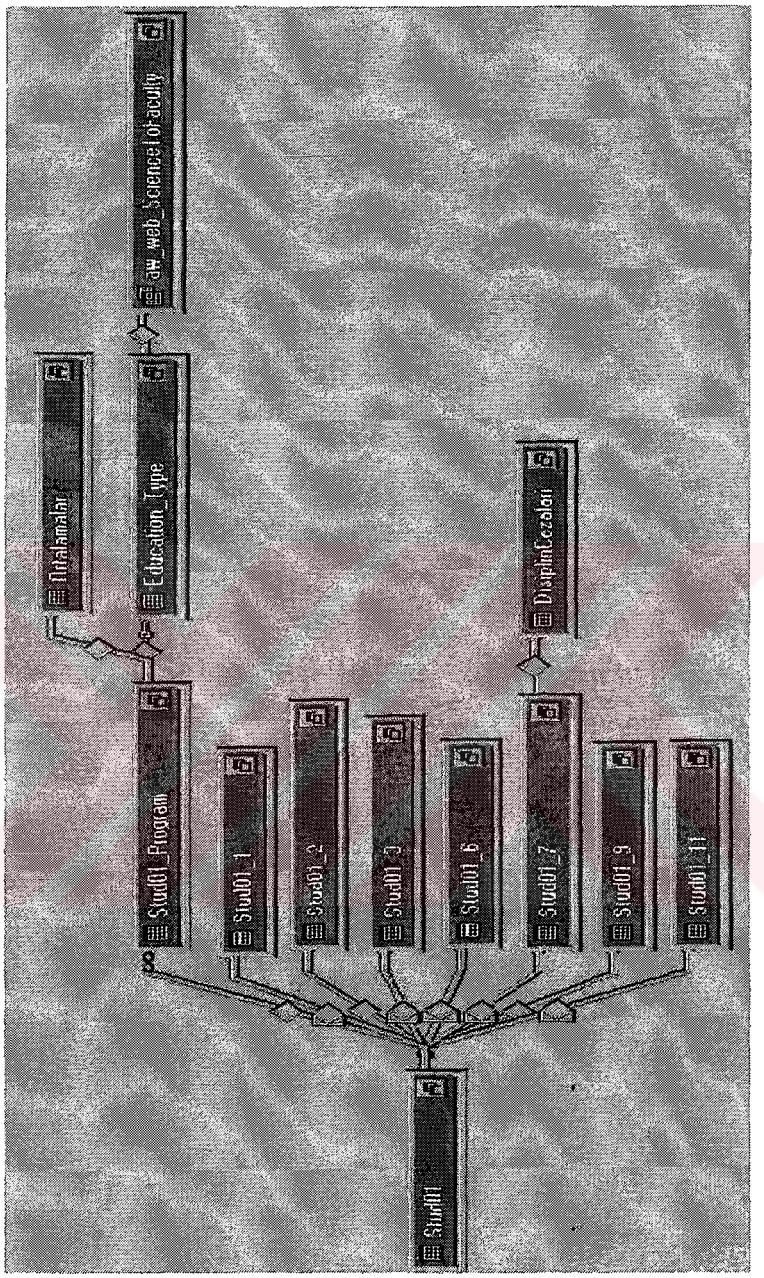


Form 7. Model oluşturma formu



Form 8. Model Denetleme formu

Ek 2. Veritabanında Kullanılan Tablolar ve Tablolar Arasındaki Bağlantılar



Ek 3. Dönüştürmede Kullanılan VBScript Kodu

```
'**********************************************************************

'                    Visual Basic Transformation Script Kodu

'**********************************************************************

Function Main()

Dim lvIL

Dim lvEVIL

Dim lvTERCSIRA

Dim lvAILEAYGE

Dim lvORTALAMA

Dim lvHEDEF

Dim dogumYili

Dim kayitYili

Dim lvYAS

'Gruplama IL_Burada tekrarlanan il adlarının bulunması veritabanında değişik şekillerde
bulunmasındandır.

If IsEksik(DTSSource("IL")) Then

' Boş Değer ise 0

lvIL = "0"

Else

lvIL = DTSSource("IL")

End If

If (lvIL = "İZMİR" Or lvIL = "İZMİR " Or lvIL = "AYDIN" Or lvIL = "AYDIN " Or lvIL = "MUĞLA"
Or lvIL = " MUĞLA" Or lvIL = "MUĞLA " Or lvIL = "MANİSA" Or lvIL = "MANİSA " Or lvIL =
"BALIKESİR" Or lvIL = "BALIKESİR " Or lvIL = "DENİZLİ" Or lvIL = "DENİZLİ " Or lvIL =
"UŞAK" Or lvIL = "AFYON" Or lvIL = "AFYON " Or lvIL = "AFYONKARAHİSAR" Or lvIL =
"KÜTAHYA" Or lvIL = "10" Or lvIL = "48" Or lvIL = "64" Or lvIL = "3" Or lvIL = "35" Or lvIL = "20"
Or lvIL = "43" Or lvIL = "45") THEN
```

```
lvIL = "EGE"

ElseIf (lvIL = "ANTALYA" Or lvIL = "MERSİN" Or lvIL = "MERSİN " Or lvIL = "İÇEL" Or lvIL =
"ADANA" Or lvIL = "ADANA " Or lvIL = "OSMANİYE" Or lvIL = "HATAY" Or lvIL = "HATAY " Or
lvIL = "İSKENDERUN" Or lvIL = "İSKENDERUN " Or lvIL = "TARSUS" Or lvIL = "ANTAKYA" Or
lvIL = "BURDUR" Or lvIL = "ISPARTA" Or lvIL = "31" Or lvIL = "80" Or lvIL = "33" Or lvIL = "7" Or
lvIL = "1") THEN

lvIL  = "AKDENIZ"

ElseIf (lvIL = "ÇANAKKALE" Or lvIL = "BURSA" Or lvIL = "BURSA " Or lvIL = "BİLECİK" Or lvIL
= "YALOVA" Or lvIL = "KOCAELİ" Or lvIL = "İZMİT" Or lvIL = "ADAPAZARI" Or lvIL =
"SAKARYA" Or lvIL = " SAKARYA" Or lvIL = "İSTANBUL" Or lvIL = "İSTANBUL " Or lvIL =
"TEKİRDAĞ" Or lvIL = " TEKİRDAĞ" Or lvIL = "TEKİRDAĞ " Or lvIL = "KIRKLARELİ" Or lvIL =
"KIRKLARELİ " Or lvIL = "EDİRNE" Or lvIL = "34" Or lvIL = "59" Or lvIL = "41" Or lvIL = "39" Or
lvIL = "54" Or lvIL = "16" Or lvIL = "17") THEN

lvIL  = "MARMARA"

ElseIf (lvIL = "ESKİŞEHİR" Or lvIL = "KONYA" Or lvIL = "KARAMAN" Or lvIL = "KARAMAN " Or
lvIL = "ANKARA" Or lvIL = "KIRIKKALE" Or lvIL = "KIRŞEHİR" Or lvIL = "NİĞDE" Or lvIL =
"NEVŞEHİR" Or lvIL = " NEVŞEHİR" Or lvIL = "AKSARAY" Or lvIL = "YOZGAT" Or lvIL = "
YOZGAT" Or lvIL = "KAYSERİ" Or lvIL = "SİVAS" Or lvIL = "38" Or lvIL = "6" Or lvIL = "70" Or
lvIL = "42") THEN

lvIL  = "ICANADOLU"

ElseIf (lvIL = "DÜZCE" Or lvIL = "BOLU" Or lvIL = "BOLU " Or lvIL = "ZONGULDAK" Or lvIL = "
ZONGULDAK" Or lvIL = "BARTIN" Or lvIL = "KARABÜK" Or lvIL = "ÇANKIRI" Or lvIL =
"KASTAMONU" Or lvIL = "SİNOP" Or lvIL = "ÇORUM" Or lvIL = "ÇORUM " Or lvIL = "AMASYA"
Or lvIL = "SAMSUN" Or lvIL = " SAMSUN" Or lvIL = "TOKAT" Or lvIL = " TOKAT" Or lvIL =
"ORDU" Or lvIL = " ORDU" Or lvIL = "ORDU " Or lvIL = "GİRESUN" Or lvIL = "GÜMÜŞHANE" Or
lvIL = "BAYBURT" Or lvIL = "TRABZON" Or lvIL = " TRABZON" Or lvIL = "RİZE" Or lvIL = "
RİZE" Or lvIL = "RİZE " Or lvIL = "ARTVİN" Or lvIL = "78" Or lvIL = "52" Or lvIL = "57" Or lvIL =
"53" Or lvIL = "28" Or lvIL = "37" Or lvIL = "60" Or lvIL = "67") THEN

lvIL  = "KARADENIZ"

ElseIf (lvIL = "MALATYA" Or lvIL = "ELAZIĞ" Or lvIL = "TUNCELİ" Or lvIL = " TUNCELİ" Or lvIL
= "ERZİNCAN" Or lvIL = "BİNGÖL" Or lvIL = "MUŞ" Or lvIL = "BİTLİS" Or lvIL = "HAKKARİ" Or
lvIL = "VAN" Or lvIL = "AĞRI" Or lvIL = "ERZURUM" Or lvIL = "KARS" Or lvIL = "IĞDIR" Or lvIL
= "ARDAHAN" Or lvIL = "24" Or lvIL = "44" Or lvIL = "36" Or lvIL = "23" Or lvIL = "25" Or lvIL =
"49") THEN

lvIL  = "DOGUANADOLU"

ElseIf (lvIL = "ŞIRNAK" Or lvIL = "SİİRT" Or lvIL = "BATMAN" Or lvIL = "MARDİN" Or lvIL =
"MARDİN " Or lvIL = "DİYARBAKIR" Or lvIL = "DİYARBAKIR " Or lvIL = "ŞANLIURFA" Or lvIL
= "Ş.URFA" Or lvIL = " ŞANLIURFA" Or lvIL = "URFA" Or lvIL = "ADIYAMAN" Or lvIL =
"GAZİANTEP" Or lvIL = "GAZİANTEP " Or lvIL = "ANTEP" Or lvIL = "KİLİS" Or lvIL =
"KAHRAMANMARAŞ" Or lvIL = "MARAŞ" Or lvIL = "21" Or lvIL = "72" Or lvIL = "2" Or lvIL =
"27" Or lvIL = "46" ) THEN
```

```
lvIL  = "GUNEYDOGU"

Else

lvIL="BELIRSIZ"

End If


'Gruplama Evil_Burada tekrarlanan il adlarının bulunması veritabanında değişik şekillerde
bulunmasındandır.

If IsEksik(DTSSource("EVIL")) Then

' Boş Değer ise 0

lvEVIL = "0"

Else

lvEVIL = DTSSource("EVIL")

End If

If (lvEVIL = "İZMİR" Or lvEVIL = "İZMİR " Or lvEVIL = "AYDIN" Or lvEVIL = "AYDIN " Or
lvEVIL = "MUĞLA" Or lvEVIL = " MUĞLA" Or lvEVIL = "MUĞLA " Or lvEVIL = "MANİSA" Or
lvEVIL = "MANİSA " Or lvEVIL = "BALIKESİR" Or lvEVIL = "BALIKESİR " Or lvEVIL =
"DENİZLİ" Or lvEVIL = "DENİZLİ " Or lvEVIL = "UŞAK" Or lvEVIL = "AFYON" Or lvEVIL =
"AFYON " Or lvEVIL = "AFYONKARAHİSAR" Or lvEVIL = "KÜTAHYA" Or lvEVIL = "10" Or
lvEVIL = "48" Or lvEVIL = "64" Or lvEVIL = "3" Or lvEVIL = "35" Or lvEVIL = "20" Or lvEVIL =
"43" Or lvEVIL = "45") THEN

lvEVIL = "EGE"

ElseIf (lvEVIL = "ANTALYA" Or lvEVIL = "MERSİN" Or lvEVIL = "MERSİN " Or lvEVIL = "İÇEL"
Or lvEVIL = "ADANA" Or lvEVIL = "ADANA " Or lvEVIL = "OSMANİYE" Or lvEVIL = "HATAY"
Or lvEVIL = "HATAY " Or lvEVIL = "İSKENDERUN" Or lvEVIL = "İSKENDERUN " Or lvEVIL =
"TARSUS" Or lvEVIL = "ANTAKYA" Or lvEVIL = "BURDUR" Or lvEVIL = "ISPARTA" Or lvEVIL
= "31" Or lvEVIL = "80" Or lvEVIL = "33" Or lvEVIL = "7" Or lvEVIL = "1") THEN

lvEVIL  = "AKDENIZ"

ElseIf (lvEVIL = "ÇANAKKALE" Or lvEVIL = "BURSA" Or lvEVIL = "BURSA " Or lvEVIL =
"BİLECİK" Or lvEVIL = "YALOVA" Or lvEVIL = "KOCAELİ" Or lvEVIL = "İZMİT" Or lvEVIL =
"ADAPAZARI" Or lvEVIL = "SAKARYA" Or lvEVIL = " SAKARYA" Or lvEVIL = "İSTANBUL" Or
lvEVIL = "İSTANBUL " Or lvEVIL = "TEKİRDAĞ" Or lvEVIL = " TEKİRDAĞ" Or lvEVIL =
"TEKİRDAĞ " Or lvEVIL = "KIRKLARELİ" Or lvEVIL = "KIRKLARELİ " Or lvEVIL = "EDİRNE"
Or lvEVIL = "34" Or lvEVIL = "59" Or lvEVIL = "41" Or lvEVIL = "39" Or lvEVIL = "54" Or lvEVIL =
"16" Or lvEVIL = "17") THEN

lvEVIL  = "MARMARA"
```

```
ElseIf (lvEVIL = "ESKİŞEHİR" Or lvEVIL = "KONYA" Or lvEVIL = "KARAMAN" Or lvEVIL =
"KARAMAN " Or lvEVIL = "ANKARA" Or lvEVIL = "KIRIKKALE" Or lvEVIL = "KIRŞEHİR" Or
lvEVIL = "NİĞDE" Or lvEVIL = "NEVŞEHİR" Or lvEVIL = " NEVŞEHİR" Or lvEVIL = "AKSARAY"
Or lvEVIL = "YOZGAT" Or lvEVIL = " YOZGAT" Or lvEVIL = "KAYSERİ" Or lvEVIL = "SİVAS" Or
lvEVIL = "38" Or lvEVIL = "6" Or lvEVIL = "70" Or lvEVIL = "42") THEN

lvEVIL  = "ICANADOLU"

ElseIf (lvEVIL = "DÜZCE" Or lvEVIL = "BOLU" Or lvEVIL = "BOLU " Or lvEVIL = "ZONGULDAK"
Or lvEVIL = " ZONGULDAK" Or lvEVIL = "BARTIN" Or lvEVIL = "KARABÜK" Or lvEVIL =
"ÇANKIRI" Or lvEVIL = "KASTAMONU" Or lvEVIL = "SİNOP" Or lvEVIL = "ÇORUM" Or lvEVIL
= "ÇORUM " Or lvEVIL = "AMASYA" Or lvEVIL = "SAMSUN" Or lvEVIL = " SAMSUN" Or lvEVIL
= "TOKAT" Or lvEVIL = " TOKAT" Or lvEVIL = "ORDU" Or lvEVIL = " ORDU" Or lvEVIL =
"ORDU " Or lvEVIL = "GİRESUN" Or lvEVIL = "GÜMÜŞHANE" Or lvEVIL = "BAYBURT" Or
lvEVIL = "TRABZON" Or lvEVIL = " TRABZON" Or lvEVIL = "RİZE" Or lvEVIL = " RİZE" Or
lvEVIL = "RİZE " Or lvEVIL = "ARTVİN" Or lvEVIL = "78" Or lvEVIL = "52" Or lvEVIL = "57" Or
lvEVIL = "53" Or lvEVIL = "28" Or lvEVIL = "37" Or lvEVIL = "60" Or lvEVIL = "67") THEN

lvEVIL  = "KARADENIZ"

ElseIf (lvEVIL = "MALATYA" Or lvEVIL = "ELAZIĞ" Or lvEVIL = "TUNCELİ" Or lvEVIL = "
TUNCELİ" Or lvEVIL = "ERZİNCAN" Or lvEVIL = "BİNGÖL" Or lvEVIL = "MUŞ" Or lvEVIL =
"BİTLİS" Or lvEVIL = "HAKKARİ" Or lvEVIL = "VAN" Or lvEVIL = "AĞRI" Or lvEVIL =
"ERZURUM" Or lvEVIL = "KARS" Or lvEVIL = "IĞDIR" Or lvEVIL = "ARDAHAN" Or lvEVIL =
"24" Or lvEVIL = "44" Or lvEVIL = "36" Or lvEVIL = "23" Or lvEVIL = "25" Or lvEVIL = "49") THEN

lvEVIL  = "DOGUANADOLU"

ElseIf (lvEVIL = "ŞIRNAK" Or lvEVIL = "SİİRT" Or lvEVIL = "BATMAN" Or lvEVIL = "MARDİN"
Or lvEVIL = "MARDİN " Or lvEVIL = "DİYARBAKIR" Or lvEVIL = "DİYARBAKIR " Or lvEVIL =
"ŞANLIURFA" Or lvEVIL = "Ş.URFA" Or lvEVIL = " ŞANLIURFA" Or lvEVIL = "URFA" Or lvEVIL
= "ADIYAMAN" Or lvEVIL = "GAZİANTEP" Or lvEVIL = "GAZİANTEP " Or lvEVIL = "ANTEP" Or
lvEVIL = "KİLİS" Or lvEVIL = "KAHRAMANMARAŞ" Or lvEVIL = "MARAŞ" Or lvEVIL = "21" Or
lvEVIL = "72" Or lvEVIL = "2" Or lvEVIL = "27" Or lvEVIL = "46" ) THEN

lvEVIL  = "GUNEYDOGU"

Else

lvEVIL="BELIRSIZ"

End If


'Gruplama lvTERCSIRA

If IsEksik(DTSSource("TERCSIRA")) Then

' Boş Değer ise 0

lvTERCSIRA = 0
```

```
Else
lvTERCSIRA = CDbl(DTSSource("TERCSIRA"))
End If
If lvTERCSIRA > 0 And lvTERCSIRA <= 5 Then
lvTERCSIRA = 1
ElseIf lvTERCSIRA > 5 And lvTERCSIRA <= 10 Then
lvTERCSIRA = 2
ElseIf lvTERCSIRA > 10 And lvTERCSIRA <= 15 Then
lvTERCSIRA = 3
ElseIf lvTERCSIRA > 15 And lvTERCSIRA <= 20 Then
lvTERCSIRA = 4
ElseIf lvTERCSIRA > 20 Then
lvTERCSIRA = 5
End If

'Gruplama AILEAYGE
If IsEksik(DTSSource("AILEAYGE")) Then
' Boş Değer ise 0
lvAILEAYGE = 0
ElseIf lvAILEAYGE > 0 And lvAILEAYGE < 1000 Then
'Gelir düzeyi sıfırları atılarak yazıldıysa
lvAILEAYGE = DTSSource("AILEAYGE")*1000000
Else
lvAILEAYGE = DTSSource("AILEAYGE")
End If
If lvAILEAYGE > 0 And lvAILEAYGE <= 200000000 Then
```

```
lvAILEAYGE = 1
ElseIf lvAILEAYGE > 200000000 And lvAILEAYGE <= 400000000 Then
lvAILEAYGE = 2
ElseIf lvAILEAYGE > 400000000 And lvAILEAYGE <= 600000000 Then
lvAILEAYGE = 3
ElseIf lvAILEAYGE > 600000000 And lvAILEAYGE <= 8000000000 Then
lvAILEAYGE = 4
ElseIf lvAILEAYGE > 8000000000 Then
lvAILEAYGE = 5
End If

'Gruplama ORTALAMA
If IsEksik(DTSSource("ORTALAMA")) Then
' Boş Değer ise 0
lvORTALAMA = 0
Else
lvORTALAMA = Csng(DTSSource("ORTALAMA"))
End If
If lvORTALAMA >= 0 And lvORTALAMA < 2 Then
lvORTALAMA = 0
lvHEDEF = 0
ElseIf lvORTALAMA >= 2 And lvORTALAMA < 3 Then
lvORTALAMA = 1
lvHEDEF = 0
ElseIf lvORTALAMA >= 3 And lvORTALAMA <=4 Then
lvORTALAMA = 1
```

```
lvHEDEF = 1
End If

'YILLAR
If IsEksik(DTSSource("DOGTARIH")) Then
'Boş değere sahip ise. -1 eksikliği gösterecektir.
dogumYili = kayitYili + 1
Else
dogumYili = cdbl(right(DTSSource("DOGTARIH"),4))
End If
kayitYili = cdbl(right(DTSSource("REGDATE"),4))
lvYAS = (kayitYili-dogumYili)

'Satır216 Gruplanmış verileri yeni tabloya yerleştirme
DTSDestination("ANAHTAR") = DTSSource("ANAHTAR")
DTSDestination("IL") = lvIL
DTSDestination("EVIL") = lvEVIL
DTSDestination("TERCSIRA") = lvTERCSIRA
DTSDestination("AILEAYGE") = lvAILEAYGE
DTSDestination("HEDEF_A") = lvORTALAMA
DTSDestination("HEDEF_B") = lvHEDEF
DTSDestination("YAS") = lvYAS
Main = DTSTransformStat_OK
End Function
```

Ek 4. 1.Model İçin Denetleme Formunda Kullanılan SQL Sorgusu

```
SELECT FLATTENED
[TEST1].[HEDEF_A] As Actual,
[DM_Tree].[HEDEF_A] As Predicted,
(SELECT
[HEDEF_A] as [Charted Value],
$Probability as Certainty
FROM
PredictHistogram([DM_Tree].[HEDEF_A])
WHERE
[HEDEF_A] = 1
)
FROM
[DM_Tree]
PREDICTION JOIN
OPENROWSET
(
'SQLOLEDB','dmserver';'sa';'www','SELECt * FROM TEST1'
) AS [TEST1]
ON
[DM_Tree].[ANAHTAR]=[TEST1].[ANAHTAR] AND
[DM_Tree].[HEDEF_B]=[TEST1].[HEDEF_B] AND
[DM_Tree].[HAZISTEK]=[TEST1].[HAZISTEK] AND
[DM_Tree].[ASKERLIK]=[TEST1].[ASKERLIK] AND
[DM_Tree].[CINSIYET]=[TEST1].[CINSIYET] AND
```

[DM_Tree].[HARCKRED]=[TEST1].[HARCKRED] AND

[DM_Tree].[EDUCTYPE]=[TEST1].[EDUCTYPE] AND

[DM_Tree].[DEPTNAME]=[TEST1].[DEPTNAME] AND

[DM_Tree].[IDCODE]=[TEST1].[IDCODE] AND

[DM_Tree].[REGTYPE]=[TEST1].[REGTYPE] AND

[DM_Tree].[OKULKOLU]=[TEST1].[OKULKOLU] AND

[DM_Tree].[HAZDILI]=[TEST1].[HAZDILI] AND

[DM_Tree].[SEMCOUNT]=[TEST1].[SEMCOUNT] AND

[DM_Tree].[YEARECNO]=[TEST1].[YEARECNO] AND

[DM_Tree].[AILEAYGE]=[TEST1].[AILEAYGE] AND

[DM_Tree].[IL2]=[TEST1].[IL2] AND

[DM_Tree].[EVIL2]=[TEST1].[EVIL2] AND

[DM_Tree].[TERCSIRA]=[TEST1].[TERCSIRA] AND

[DM_Tree].[TERCIH2]=[TEST1].[TERCIH2]

Ek 5. Çalışmada Kullanılan Sütunlar

Çalışmada Kullanılan İsim	Orijinal İsim	İçerdiği Veri
ACIKLAMA*	Aciklama	Ceza alma nedeni ve alınan ceza türü
AILEAYGE	AileAylikGelir	Ailenin aylık geliri
AILEKISI	AileKisiSayisi	Ailedeki kişi sayısı
AILESNO*	AileSiraNo	Aile sıra no (nüfus cüzdanı bilgisi)
ANAHTAR	Stud01RecordNo	Kayıt numarası **(Anahtar sütun)**
ANNEADI*	AnaAdi	Anne adı
ANNEMES	AnneMeslek	Anne mesleği
ANNESOSD	AnneSosyalDurum	Anne sosyal durumu
ANNEYASA	AnneYasam	Anne sağ olup olmadığı
ARABANOT	ArabaNot	Araba var-yok
ARABASAY	ArabaSayisi	Araba sayısı
ASKERLIK	AskerlikDurumu	Askerlik durumu
ASKSUBE*	AskerlikSubesi	Askerlik şubesi
ATTEND***	Attendance	Devam etme (zorunlu)
BABAADI*	BabaAdi	Baba adı
BABABYKS	BabaBakYukKisiSayisi	Babanın bakmakla yükümlü olduğu kişi s.
BABAMES	BabaMeslek	Baba mesleği
BABASOSD	BabaSosyalDurum	Babanın sosyal durumu
BABAYASA	BabaYasam	Baba sağ olup olmadığı
CAPACITY*	Capacity	Kapasite (tamamen boş değer içeriyor)
CEZA	Ceza	Eğitim boyunca aldığı ceza
CILTNO*	CiltNo	Cilt no (nüfus cüzdanı bilgisi)
CINSIYET	Cinsiyet	Cinsiyet
DEPTNAME♣	DeptName	Bölüm adı
DGSTERCS	DGSTercihSirasi	DGS tercih sırası
DGSYPUAN	DGSYPuani	DGS puanı
DINI	Dini	Dini
DIPDEREC*	DiplomaDerecesi	Lise diploma derecesi
DIPNOTU	DiplomaNotu	Lise diploma notu
DOGTARIH	DogumTarihi	Doğum tarihi
DOGYERI**	DogumYeri	Doğum yeri
DOGYERIL**	DogumYerili	Doğum yeri sadece il
DREGDATE	DeRegistrationDate	Okuldan ayrılma tarihi
EDUCTYPE	EducationType	Eğitim türü (Normal / II. Öğretim)
EVADRESI*	EvAdresi	Ev adresi
EVDURUMU	EvDurumu	Ev sahibi olup olunmadığı
EVIL♣	Evil	Evin bulunduğu il
EVILCE*	Evlce	Evin bulunduğu ilçe
EVPOSKOD**	EvPostaKodu	Ev posta kodu
EVULKE	EvUlke	Evin bulunduğu ülke
EXPR1*	Expr1	Bilinmiyor

Çalışmada Kullanılan İsim	Orijinal İsim	İçerdiği Veri
FACNAME***	FacultyName	Fakülte adı (İİBF)
FACTYPE***	FacultyType	Fakülte türü (Fakülte)
GELIRKAY	GelirKaynaklari	Gelir kaynakları
GMENKIGE	GayrimenkulKiraGeliri	Gayri menkulden gelen kire geliri
GMENNOT	GayrimenkulNot	Gayri menkul var-yok
GMENSAY	GayrimenkulSayisi	Gayri menkul sayısı
HARCKREDI	HarcKredisi	Harç kredisi alıp almadığı
HAZBATAR*	HazBasTarihi	Hazırlığa başlama tarihi
HAZBITAR*	HazBitTarihi	Hazırlık bitirme tarihi
HAZDEREC	HazDerecesi	Hazırlık bitirme derecesi
HAZDILI	HazDili	Hazırlık dili
HAZIRLIK	Hazirlik	Hazırlık alıp almadığı
HAZISTEK*	HazirlikIstemi	Hazırlık isteği
HAZSINAV*	HazSinavi	Hazırlık sınav sonucu
IDCODE	IDCode	Bölüm ve eğitim türünü birlikte tutan kod
IL♣	Il	Nufusa kayıtlı olduğu il
ILCE*	Ilce	Nufusa kayıtlı olduğu ilçe
INFAZTAR*	InfazTarihi	Ceza infaz tarihi
INSTYEAR***	InstructionYear	Eğitim yılı (4)
KALILCE*	KaldigiIlce	Kaldığı ilçe
KALSEHIR**	KaldigiSehir	Kaldığı şehir
KANGRUP*	KanGrubu	Kan grubu
KARARTAR*	KararTarihi	Ceza karar tarihi
KONTUR***	KontenjanTuru	Kontenjan türü (Genel)
MAHKOY*	MahalleKoy	Mahalle-köy (nüfus cüzdanı bilgisi)
MEDENHAL	MedeniHali	Medeni hal
MEZDEREC	MezuniyetDerecesi	Mezuniyet derecesi
MEZYILI	MezuniyetYili	Önlisans mezuniyet yılı
OGRAYGE	OgrenciAylikGelir	Öğrencinin aylık geliri
OGREHLIY*	OgrenciEhliyet	Öğrenci ehliyet sahibi veya değil
OGRGEKAY	OgrenciGelirKaynagi	Öğrenci gelir kaynağı
OKULADI*	OkulAdi	Bitirdiği lisenin adı
OKULAGIR	OkulaGirisTarihi	üniversiteye giriş tarihi
OKULBIR	OkulBirincisi	Lisede okul birincisi olup olmadığı
OKULKOLU	OkulKolu	Lise okul türü
OKULTURU	OkulTuru	Bitirdiği lisedeki okul türü
OKUMATUR	OkumaTuru	Önlisans okuma türü
OLAKAORT	OLAkademikOrtalama	Akademik ortalama
OLBIYILI	OLBitirmeYili	Bitirme yılı
ONLISMEZ	OnlisansMezuniyetAlani	Önlisans mezunu ise alanı
ORTALAMA	Ort	Genel başarı ortalaması **(Hedef sütun)**
OSS_EW	OSS_EWGrade	ÖSS eşit ağırlık puanı
OSS_M	OSS_MGrade	ÖSS sayısal puanı
OSS_S	OSS_SGrade	ÖSS sözel puanı
OSSDATE	OSSDate	ÖSS sınav tarihi

Çalışmada Kullanılan İsim	Orijinal İsim	İçerdiği Veri
OSSPUTUR***	OSS_PuanTuru	ÖSS puan türü (Eşit ağırlık)
OYSGRADE	OYSGrade	ÖYS puanı
PROGTYPE	ProgramType	Program türü
REGDATE	RegistrationDate	Okula kayıt tarihi
REGNOTE	RegistrationNote	Kayıt notu
REGTYPE	RegistrationType	Kayıt türü (Normal veya değil)
SCINAME**	ScienceName	Bölüm adı
SEMCOUNT	SemestreCount	Bulunduğu yarıyıl toplamı
SEMRECNO	EduSemRecordNo	Üniversitede kaçıncı yarıyılda bulunduğu
SIRANO*	SiraNo	Sıra no (nüfus cüzdanı bilgisi)
STATE***	State	Durum (Kayıt yenileme)
STATUS*	Status	Aktif olarak eğitim alıp almadığı
STUDNAME*	StudentName	Adı
STUDNO**	StudentNumber	Öğrenci numarası (Anahtar olabilir)
STUDYEAR**	StudentYear	Okulda bulunduğu yıl
SUSTATUS*	SubStatus	Mezun veya değil
TERCSIRA	TercihSirasi	Bulunduğu bölümü tercih sırası
UYRUK	Uyruk	Uyruk
VERILNED*	VerilisNedeni	Veriliş nedeni (nüfus cüzdanı bilgisi)
VERILTAR*	VerildigiTarih	Verildiği tarih (nüfus cüzdanı bilgisi)
VERILYER*	VerildigiYer	Nüfus cüzdanının verildiği yer
YABDIL	YabanciDil	Yabancı dili
YAKSODUR	YakininSosyalDurumu	Yakınının (akrabasının) sosyal durumu
YEARECNO♣	EduYearRecordNo	Üniversitede kaçıncı yılında bulunduğu
YEARUPLI***	YearUpperLimit	Okuma üst sınırı (7)

ÖZGEÇMİŞ

06.02.1976 yılında İzmir'de doğdu. İlk ve orta öğrenimini Akıncılar İlkokulu, Şirinyer Lisesi'nin ortaokul kısmı, Çınarlı Endüstri Meslek Lisesi'nin teknik lise elektronik bölümünde tamamladı. Marmara Üniversitesi, Teknik Eğitim Fakültesi, Elektronik Bilgisayar Eğitimi Bölümü'nden 1999 yılında mezun oldu. 2000–2002 yılları arasında Fethiye Endüstri Meslek Lisesi'nde Elektronik Öğretmeni olarak görev yaptı. 2002 yılından itibaren Muğla Üniversitesi, Teknik Eğitim Fakültesi, Elektronik ve Bilgisayar Eğitimi Bölümü'nde Öğretim Görevlisi olarak görev yapmaktadır. Yabancı dil olarak İngilizce bilmektedir. Beşi uluslararası olmak üzere altı adet yayımlanmış konferans bildirisi bulunmaktadır.