

**T.C.**  
**GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ**  
**MÜHENDİSLİK VE FEN BİLİMLERİ**  
**ENSTİTÜSÜ**

**ROTALAMA PROBLEMİNİN VERİ**  
**MADENCİLİĞİ İLE ÇÖZÜMÜ**

**Derya ŞAHİN**  
**YÜKSEK LİSANS**  
**BİLGİSAYAR MÜHENDİSLİĞİ**  
**ANABİLİM DALI**

**Gebze**  
**2010**

**T.C.**  
**GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ**  
**MÜHENDİSLİK VE FEN BİLİMLERİ**  
**ENSTİTÜSÜ**

**ROTALAMA PROBLEMİNİN VERİ**  
**MADENCİLİĞİ İLE ÇÖZÜMÜ**

**Derya ŞAHİN**  
**YÜKSEK LİSANS**  
**BİLGİSAYAR MÜHENDİSLİĞİ**  
**ANABİLİM DALI**

**TEZ DANIŞMANI**  
**Dr. Hidayet TAKÇI**

**Gebze**  
**2010**



## YÜKSEK LİSANS TEZİ JÜRİ ONAY SAYFASI

G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 07/10/2010 tarih ve 210/25... sayılı kararıyla oluşturulan jüri tarafından 07/07/2010... tarihinde tez savunma sınavı yapılan Deniz SAHİN.....'in tez çalışması Bilgisayar Mühendisliği Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

### JÜRİ

ÜYE

(TEZ DANIŞMANI) :

Dr. Hidayet TAKCI.....

ÜYE

:

Yrd. Doç. Dr. Mehmet GÖKTÜRK

ÜYE

:

Yrd. Doç. Dr. İlyas KANDEMİR

### ONAY

G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ...../...../20... tarih ve ...../..... sayılı kararı.

İMZA/MÜHÜR

## ÖZET

### **TEZ BAŞLIĞI: ROTALAMA PROBLEMİNİN VERİ MADENCİLİĞİ İLE ÇÖZÜMÜ**

**YAZAR ADI: DERYA ŞAHİN**

Rotalama, başta bilgisayar ağları olmak üzere; aktarımın söz konusu olduğu birçok ortamda önemli bir optimizasyon problemidir. Konunun önemi maliyet sebebiyledir. Doğru rotalar maliyet düşürücü yanlış rotalar ise maliyet artırıcı etkiye sahiptir. Bu çalışmanın amacı Araç Rotalama probleminin veri madenciliği teknikleri ile çözümünü sağlamaktır. Bu çalışmada; bir otomobil firmasında üretilen araçların bayilere en az maliyetle ulaştırılması için bir sistem önerilmiştir.

Otomobil firmasında üretilen araçlar ülkemizin çeşitli bölgelerindeki bayilere tırlar yardımıyla ulaşmaktadır. Bir tırı dolduran nakliyalarda rota bellidir ancak bir tırı doldurmayan nakliyatlar için bir tır birden fazla bayiye uğrayabilmektedir. Hangi bayilerin aynı rotada olabileceği bilgisi tarihsel veriden, veri madenciliği teknikleriyle ortaya çıkarılacaktır. Çalışmamızda yapılan işlerden biri; nakliyat rotalarından en sık kullanılanların bulunmasıdır. Bu çözümün ilk aşamasıdır. İkinci aşamada ise bu rotaları kullanarak en uygun rotayı belirleyen model oluşturulacaktır.

## **SUMMARY**

### **THESIS TITLE: DATA MINING SOLUTION FOR VEHICLE ROUTING PROBLEM**

### **THESIS AUTHOR: DERYA ŞAHİN**

Routing is an important optimization subject for transportation. The importance is about cost. Right routes have the fact of cost reduction meanwhile wrong routes have the fact of cost increment. The reason of this study is the solution of vehicle routing problem with data mining techniques. A system is proposed for transportation of vehicles produced from factory to vendors with minimum cost. Vehicles are transported to vendors of all the country by transporters. If all the vehicles in a transporter are for the same vendor, the route is definite. But if there are not enough cars for same vendor, a transporter travels more than one vendor. We can explore that which vendors could be in same route by mining historical data. In our study we explore which routes are used frequently. This is the first phase of the solution. In second phase a model is constructed which finds best route from frequent used routes.

## TEŞEKKÜR

Yüksek lisans eğitimim ile tez çalışmam boyunca bana destek veren ve teşvik eden tez danışmanım Gebze Yüksek Teknoloji Enstitüsü Bilgisayar Mühendisliği Bölümü Öğretim Üyesi Sayın Dr. Hidayet TAKÇI' ya,

Sevkiyat Sistemi ve bayi araç rotalaması hakkındaki süreç bilgisini esirgemeyen ve bu çalışmanın ortaya çıkmasında katkısı büyük olan tüm sevkiyat sistemi çalışanlarına,

Tez yazım sürecinde desteğini yanımda bulduğum tüm arkadaşlarıma,

Bugünlere gelmemde büyük emeği olan, hayatım boyunca maddi ve manevi destekleri ile her zaman yanımda olan aileme, özellikle tüm eğitim hayatım boyunca hep bana destek ve yardımcı olan Anneme en içten dileklerle teşekkür ederim.

# İÇİNDEKİLER DİZİNİ

ÖZET	iv
SUMMARY	v
TEŞEKKÜR	vi
SİMGELER VE KISALTMALAR DİZİNİ	ix
ŞEKİLLER DİZİNİ	x
ÇİZELGELER DİZİNİ	xi
1. GİRİŞ	1
1.1. Problem Detayı	1
1.2. Çalışmanın Çözüm Yöntemi	3
1.3. Çalışmanın Planı	4
2. VERİ MADENCİLİĞİ	6
2.1. Veri Madenciliği Nedir?	6
2.2. Veri Madenciliğinin Uygulama Alanları	8
2.3. Veri Madenciliği Model ve Teknikleri	10
2.3.1. Sınıflandırma (Classification)	10
2.3.2. Kestirim (Estimation)	11
2.3.3. Tahmin (Prediction)	12
2.3.4. Benzer Gruplama (Affinity Grouping) /Birliktelik Analizi	13
2.3.5. Kümeleme (Clustering)	13
2.3.6. Tanımlama ve profil oluşturma (Description and Profiling)	14
3. Birliktelik Analizi ve Apriori Algoritması	15
3.1. Birliktelik Kuralı	15
3.2. Apriori Algoritmasının Yapısı	16
4. ARAÇ ROTALAMA	20
4.1. Lojistik	20
4.1.1. Lojistik Nedir?	20
4.1.2. Lojistik Yönetiminin Önemi?	21
4.1.3. Lojistik Sistemindeki Problemler	23
4.1.3.1. Araç Filosu Yönetim	23
4.1.3.2. Araç Rotalama	23
4.1.4. Lojistik Problemlerini Modelleme	24
4.2. ARP Çeşitleri	25

4.3. ARP İin özüm Yöntemleri	26
4.4. ARP özümünde Veri Madencilięi Kullanımı	28
5. ARP' nin Veri Madencilięi Kullanarak özümü	29
5.1. SRK Oluşturma	30
5.2. Rotalama	35
6.SONU	39
KAYNAKLAR	41
ÖZGEMİŐ	44

## SİMGELER VE KISALTMALAR DİZİNİ

ARP	: Araç Rotalama Problemi
SRK	: Sık kullanılan Rotalar Kümesi
KDD	: Knowledge Discovery From Databases
VTBK	: Veri Tabanlarında Bilgi Keşfi
$\min_{des}$	: Minimum Destek Değeri
ARM	: Araç Rotalama Modülü

## ŞEKİLLER DİZİNİ

<b><u>Sekil</u></b>	<b><u>Sayfa</u></b>
2.1. Veri Tabanlarında Bilgi Keşif Süreci	7
3.1. Klasik Apriori Algoritması Özet Kodu	17
3.2. Apriori-Gen Fonksiyonu	18
4.1. Ağ [Simchi-Levi and Bramel, 1997]	20
4.2. Araç Rotalama Problemi Çözüm Yöntemleri	27
5.1. Sistem Modeli	29
5.2. Sistemi Giriş Sayfası	30
5.3. İki ile Uğrayan Rotalar ve Sıklığı	33
5.4. Üç ile Uğrayan Rotalar ve Sıklığı	34
5.5. Rotalama Algoritması	36
5.6. Rotalama Sonucu	38

## ÇİZELGELER DİZİNİ

<b><u>Cizelge</u></b>	<b><u>Sayfa</u></b>
5.1. İllerin Başka İller ile Gönderimi Oranı	32
5.2. Rota Bulma Süresi	35
5.3. Model Değişkenleri ve Anlamları	36

# 1.GİRİŞ

Son zamanlarda bilgi teknolojilerinde yaşanan dikkat çekici gelişmeler sayesinde depolanan veri boyutları çok büyük hacimlere ulaşmıştır. Bilgisayar sistemlerinin bir taraftan ucuzlaması diğer taraftan ise işlem güçlerinin artması veri artışında etkili olmuştur. İşlemciler gittikçe hızlanmış, disklerin kapasiteleri artmıştır. Böylece artık bilgisayarlar daha büyük miktardaki veriyi saklayabilmekte ve daha kısa sürede işleyebilmektedirler.

Sürekli olarak artan veri yığınları belli bir amaca yönelik olarak islenip bilgiye dönüştürülmediği sürece bizim için değersizdir. Onu anlamlı ve değerli hale getirebilmek için veri analiz araçları önemli rol oynamaktadır. Veri islenip bilgiye dönüştürüldüğünde bizim için bir anlam ifade etmeye baslar. Büyük veri yığınlarından yararlı bilgiye erişim ihtiyarcını karşılamak için kullanılan veri analiz araçlarından biri veri madenciliğidir. Veri madenciliği son zamanlarda giderek önemi artan bir araştırma alanı haline gelmiştir. Birçok tanımı olmakla birlikte veri madenciliği için en bilinen tanıma göre; Veri madenciliği; veri içinden ilginç, üstü kapalı ve anlamlı örüntüleri otomatik veya yarı otomatik olarak bulma işlemidir [Ayre, 2006].

Günümüzün yoğun rekabet ortamında, kısa yaşam eğrisine sahip ürünler ve müşterilerin artan beklentileri, üreticileri; dağıtım sistemlerine yatırım yapmalarına ve gereken önemi vermelerine zorlamaktadır. Bu durum, iletişim ve ulaşım teknolojilerindeki değişimle birlikte, örneğin mobil iletişim ve günaşırı dağıtım gibi, lojistik yönetiminin sürekli gelişimine neden olmuştur. Lojistik sisteminin en önemli parçası ve problemi ise **Araç Rotalama Problemidir**.

Araç rotalama problemi (ARP), bir veya birkaç depoya sahip ve belirli sayıda müşterisi olan firmaların, ürün dağıtımını minimum mesafe ile gerçekleştirmesi problemidir. Genellikle bu probleme araç kapasitesi, yol uzunluğu gibi bazı kısıtlar eklenir [Simchi-Levi and Bramel, 1997].

## 1.1. Problem Detayı

Ürün dağıtımını yapan firmalar için önemli problemlerden birisi araç rotalama problemidir. Doğru yapılmadığı takdirde maliyet artırıcı bir yönü olan bir problemidir. Çünkü hatalı rotalamalar fazla maliyet anlamına gelmektedir. Dağıtım

maliyetleri, ürün maliyetlerinin yaklaşık %15-20'sini oluşturmaktadır [Rushton et al, 2006]. Genellikle firmalar bu probleme herhangi bir matematiksel model kullanmadan, geçmiş tecrübelerinden veya kendi geliştirdikleri algoritmaları kullanarak çözüm aramaktadırlar. Firmaların bir matematiksel model kullanmadan yaptığı bu araç rotaları, yüksek dağıtım maliyetlerinin ortaya çıkmasına neden olmaktadır. Etkin olmayan dağıtım rotaları, firmalara ekstra bir maliyet yaratmaktadır. Sonuç olarak etkin bir dağıtım rotası oluşturmak firmaya büyük bir tasarruf oluşturacak ve günümüzdeki rekabetçi ortamda önemli bir avantaj sağlayacaktır.

Bu çalışmada problem bir otomobil firmasında üretilen araçların stok alanından yurt içi bayilere nakledilmesi olayıdır. Nakliyat belirli bir kapasitesi olan tırlar ile gerçekleştirilecektir. Bayilerin sipariş adetlerine göre tırlar oluşturulmaktadır. Kimi zaman bir bayinin aynı anda bir tırı dolduracak kadar siparişi bulunmayabilir, bu durumda birbirine benzeyen (coğrafi yakınlık, aynı yol üzerinde bulunma) bayi araçları tek bir tır ile gönderilebilir. Hangi bayilere taşınan araçların aynı tırda bulunabileceği bilgisi herhangi bir sistemde bulunmamaktadır. Rotaları iyi bilen bir kullanıcı aynı tırda gidebilecek bayileri seçer ve ona göre tır yükü oluşturulur. İşlem insana bağımlı olup bunun yerine bu işlemin otomatik yöntemlerle ve kabul edilebilir bir hızda yapılması gerekliliği vardır. Bu çalışmada yukarıda anlatılan probleme veri madenciliği teknikleri ile çözüm aranmaktadır.

ARP, En basit haliyle bile NP-Hard problemler arasında gösterilir. Lojistik ve dağıtım konularında popüler olduğundan 40 yıldır üzerinde çalışılan bir konudur. Bu güne kadar ARP problemlerine çözüm olarak birçok metot ve algoritma geliştirilmiştir. Genel olarak çözüm için sunulan ARP yaklaşımları, kesin yöntem (exact methods), sezgisel yöntem (heuristics) ve meta-sezgisel (meta heuristics) yöntem olarak sınıflandırılabilir.

Araç Rotalama Problemlerinde kullanılan kesin yöntem algoritmaları, orta büyüklükteki problemleri çözmek için çok büyük bilgisayar sürelerine ihtiyaç duyduğu için fazla tatmin edici değildir. Sezgisel yöntemler ise kısa sürede optimuma yakın sonuçlar vermektedir. Günlük hayatta karşılaşılan problemler çoğunlukla çok büyük boyutta ve karmaşık olduğundan, bu tür problemleri çözmek için sezgisel yöntemler daha uygundur. Literatürde de ARP için az sayıda kesin

çözüm yöntemi kullanılarak yapılmış çalışmalar vardır. Genellikle araştırmacılar sezgisel çözüm yöntemi üzerinde çalışmalarını yoğunlaştırmışlardır.

## 1.2. Çalışmanın Çözüm Yöntemi

Problem incelendiğinde aynı tırla araç gönderilen bayilerin çok sık değişmediği gözlemlenmiştir. Şu ana kadar gönderilen araçların rotaları bundan sonraki sevkiyatlar için de kullanılması en yüksek olasılığa sahip olan rotalardır. Şu ana kadar kullanılan rotalar ve kullanım sıklıkları birliktelik analizi yöntemiyle var olan veriden çıkarılabilir. Sıklığı belirli bir eşik değerini aşan rotalar ile sık kullanılan rotalar kümesi (SRK) oluşturulabilir ve bu rotanın kullanım sıklığı o rotanın değerini verir. Bu çözümün ilk aşamasıdır. İkinci aşamada park alanındaki araçlar 1. Aşamada bulunmuş rotalara ve kurallara göre değerini maksimize edecek şekilde tır kapasitesine göre gruplara ayrılır. Çözüm iki aşamalı olarak tasarlanmıştır. İlk aşama yani SRK oluşturma adımı her rotalama işleminde yenilenmesi gerekmediğinden ihtiyaca göre haftalık ya da aylık periyotlarda oluşturulabilir. İkinci aşama, tır rotalarının belirlenmesi ise park alanında bekleyen araç sayısına göre günde en az 1-2 kez yapılabilecek bir işlemdir. Bu yüzden kabul edilebilir bir zamanda çözüm üretmesi beklenir.

Rotalama problemine sadece veri madenciliği teknikleriyle çözüm bulma fikri aslında problemin doğasından kaynaklanmaktadır. Literatürde farklı tür rotalama problemleri ve çözümleri bulunmaktadır. Ancak rotaların çok da sık değişmediği durumlarda şu ana kadar kullanılmış rotalara göre bundan sonraki rotaları da çıkarmak klasik yöntemlerdeki gibi rotaların tek tek denendiği kombinasyonel çözümden daha etkin olacaktır.

Çözümün ilk aşaması olarak önerdiğimiz SRK bulma adımında Apriori algoritması gerçekleştirilerek sık kullanılan rotalar ortaya çıkarılacaktır [Agrawal and Srikant, 1994],[ Agrawal and Srikant, 1995]. Aslında birliktelik analizi yapmak için Apriori algoritmasının daha gelişmiş sürümleri mevcuttur ama inceleyeceğimiz küme çok büyük olmadığından ve SRK bulma adımı çok sık çağrılmadığından daha kolay bir sürüm gerçekleşmiştir. Bu algorithmada aynı rota üzerinde bulunan merkezler veritabanı taranarak bulunur ve minimum destek değerine ulaşan kayıtlar sık kullanılan rota olarak değerlendirilir. Minimum destek değeri ( $min_{des}$ ) o rotanın sık kullanılan rota olarak işaretlenmesi için ulaşması gereken tekrarlanma sayısıdır. İlk

aşama sık kullanılan rotaların ortaya çıkarıldığı, aynı zamanda verilerin de incelenip karakteristiklerinin ortaya çıkarıldığı adım olmuştur. İkinci aşamada elimizde tıra yüklenmesi gereken araçlar vardır. Bu araçların tır kapasitesine göre kümelere, benzerliklerine ve bazı kurallara göre ayrılması gerekmektedir. Benzerlik değeri bayiler aynı ildeyse ve iki bayi çok sık aynı rotada bulunuyorsa maksimumdur. Bu benzerlik değeri ve kurallara göre tırlar ve her bir tırın değeri oluşur. Bu değerlerin maksimum olduğu yükler oluşturulur ve kalan araçlar ile algoritma birkaç kez daha çalıştırılarak olası en iyi yükler oluşturulur.

Ortaya koyduğumuz algoritma ile özellikle sık değişmeyen araç rotalarını bulmak için veri madenciliği kullanımı örneklendirilmiştir. Geliştirilen model, klasik kombinyonel rotalama problemlerine de baktığımızda bayi sayısı az olmamasına rağmen kabul edilebilir zamanlarda sonuç üretmektedir. Önerimiz şu ana kadar yapılmış olan rotalama algoritmalarına alternatif olarak gösterilebilir. Yöntemimiz için en önemli konu veri madenciliği yapabilmek için rotalama ile ilgili geçmiş verinin bulunmasıdır.

Şu ana kadar veri madenciliği teknikleri sezgisel yöntemlerin iyileştirilmesinde kullanılmıştır. Örneğin, kümeleme analizi sayesinde müşteriler benzerliklerine (coğrafi yakınlık vs.) göre gruplara ayrılabilir ve bu gruplara göre rotalama yapılabilir [Hu and Huang,2007]. Ya da müşteri talebinin bilinmediği durumda olası talep şu ana kadar olan talep verisinden elde edilebilir [Marković et al., 2005]. Genetik algoritmalarındaki deneme sayısını azaltmak için en iyi popülasyonlarda sık kullanılan rotalar birliktelik analizi ve sınıflandırma ile belirlenir [Santos et. Al, 2006]. Önerdiğimiz çözüm ise ARP problemi için sadece veri madenciliği tekniklerinin kullanıldığı bir çözümdür.

Çalışmada veri tabanı olarak Oracle, geliştirme dili olarak Oracle PL/SQL önyüz tasarımında ise Oracle APEX kullanılmıştır.

### **1.3. Çalışmanın Planı**

Tez çalışması beş bölümden oluşmaktadır. Tezin ikinci bölümünde veri madenciliği hakkında genel bilgiler verilerek giriş yapılmış ardından Veri Tabanlarında Bilgi Keşfi (VTBK) adımları anlatılmıştır. Sonrasında veri madenciliğinin uygulama alanlarına değinilmiştir. Veri madenciliği modelleme teknikleri sırayla ele alınıp açıklanmıştır.

Üçüncü bölümde veri madenciliği tekniklerinden tezde de kullandığımız Birliktelik Kuralları Analizi detaylı olarak incelenmiştir. Birliktelik Kuralları Analizi tanımı, özelliklerine yer verilmiştir. Birliktelik Analizinin hangi durumlarda kullanılabileceği açıklanmıştır. Birliktelik Analizine ilişkin olarak geliştirilmiş algoritmalar incelenmiş en popüler olanı ve gerçekleştirimde kullandığımız Apriori algoritması detaylarıyla anlatılmıştır.

Dördüncü bölümde başlangıçta Araç Rotalama Problemi (ARP) hakkında genel bilgi verilmiş sonrasında ARP türleri açıklanmıştır. Bir sonraki adımda şu ana kadar geliştirilmiş tüm yaklaşımlar ve çözüm yolları hakkında bilgi verilmiştir. Ardından ARP problemlerinin çözümünde veri madenciliği kullanımı örneklendirilmiştir.

Beşinci bölümde, ilk aşamada elimizdeki problem, inceleyeceğimiz veri yapısı ve ulaşmak istediğimiz sonuç açıklanmıştır. Ardından sadece veri madenciliği teknikleri kullanarak ARP probleminin nasıl bir çözüm modeli oluşturulduğu açıklanmıştır. Çözüm gerçekleştirim detayları anlatılmıştır.

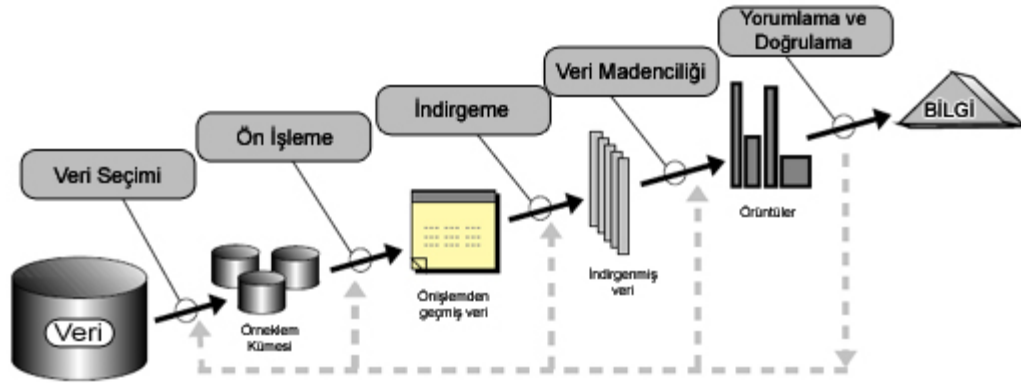
Son bölümde geliştirilen çözümün örnekleme üzerindeki çalışma süreleri ve var olan yöntemlere olan üstünlüklerinden bahsedilmiştir.

## 2. VERİ MADENCİLİĞİ

### 2.1. Veri Madenciliği Nedir?

Veri madenciliği veri içinden ilginç, üstü kapalı ve anlamlı örüntüleri otomatik veya yarı otomatik olarak bulma işlemidir [Ayre, 2006]. Veri Madenciliği sihir değildir. Yıllardır, istatistikçiler veri tabanlarını elle kazımakta, istatistiksel açıdan önemli ilişkiler aramaktadır. Veri Madenciliği, bu süreci otomatik olarak gerçekleştirmektedir. Veri Madenciliği veri kümesi içerisinde keşfedilmemiş örüntüleri bulmayı hedefleyen teknikler koleksiyonunu betimlemektedir. Veri Madenciliğinin amacı, geçmiş faaliyetlerin analizini temel alarak gelecekteki davranışların tahminine yönelik karar-verme modelleri yaratmaktır. Frawley veri madenciliğini: “Daha önceden bilinmeyen ve potansiyel olarak yararlı olma ihtimaline sahip verinin keşfedilmesi” olarak tanımlamıştır. Berry ve Linoff bu kavrama “Anlamlı kuralların ve örüntülerin bulunması için geniş veri yığınları üzerine yapılan keşif ve analiz işlemleri” şeklinde bir açıklama getirirken, Sever ve Oğuz çalışmalarında veri madenciliği hakkında “Önceden bilinmeyen, veri içinde gizli, anlamlı ve yararlı örüntülerin büyük ölçekli veritabanlarından otomatik biçimde elde edilmesini sağlayan veri tabanlarında bilgi keşfi süreci içerisinde bir adımdır.” tanımını kullanmışlardır [Öğüt].

Veri madenciliği deyimi yanlış kullanılan bir deyim olabileceğinden bilim adamları tarafından buna eş değer baksa adlandırmalarda literatüre geçmiştir. Bunlar; Veritabanlarında bilgi keşfi (VTBK) (KDD Knowledge Discovery in Databases), Bilgi Çıkarımı (Knowledge Extraction), Veri ve Örüntü analizi (Data/ Pattern Analysis), Veri Tarama (Data Dredging), Bilgi Keşfi (Knowledge Discovery), Veri Arkeolojisi (Data Archaeology), Veri Avcılığı (Data Fishing), Bilgi Üretimi (Knowledge Creation) ve Bilgi Hasadıdır (Information Harvesting). Bu adlandırmalardan veri madenciliği deyimi yerine en çok kullanılanı VTBK olmuştur. Bununla birlikte aslında VTBK, veriler arasından yararlı bilgileri keşfetme sürecidir ve veri madenciliği VTBK sürecinin önemli bir adımıdır. VTBK süreci ise veri hazırlama, veri seçme, veri temizleme ve veri madenciliği sonucu çıkan sonuçların yorumlanması gibi ek adımlarla birlikte veriden türetilen yararlı bilginin elde edilmesi demektir. Şekil 2.1 VTBK adımlarını göstermektedir. Bu adımlar:



Şekil 2.1. Veri Tabanlarında Bilgi Keşif Süreci[Öğüt]

Veri madenciliği, VTBK işleminin adımlarından biridir. VTBK işleminin adımları farklı kişiler tarafından farklı modellerle ifade edilmiştir. Han tarafından sunulan modelde VTBK işlemi yedi adımda meydana gelmektedir. Bu adımlar sırasıyla veri temizleme, veri birleştirme, veri seçme, veri dönüştürme, veri madenciliği, örüntü değerlendirme ve bilgi sunumudur. VTBK sürecindeki adımlar Şekil 2.1.' de görsel olarak ifade edilmiştir. Han'ın sunduğu modeldeki VTBK sürecinde yer alan adımlar açıklamaları ile aşağıda belirtilmiştir [Han and Kamber, 2001]:

1) Veri Temizleme (Data Cleaning): Gerçek hayatta kullanılan veritabanları içindeki veriler kirlenmeye, eksik hale gelmeye ve tutarsız olmaya eğilimlidirler. Bu nedenle verilerin kullanılmadan önce bazı ön işlemlerden geçmeleri gerekir. Ön işlemlerden geçen veriler üzerinde veri madenciliği algoritmalarının uygulanması ile daha kaliteli sonuçlar elde edilir. Bu ön işlemlerden biri veri temizlemedir. Veri temizleme ile veritabanlarındaki eksik, tutarsız ve gürültülü veriler giderilir.

2) Veri Birleştirme (Data Integration): Veri temizlemeden sonra veri birleştirme işlemi uygulanır. Veri birleştirme, çeşitli kaynaklardan gelen verilerin tek bir veri ambarı altında toplanmasıdır.

3) Veri Seçme (Data Selection): Veritabanlarında üzerinde işlem yapılacak olan veri seçilir ve veri türleri bu aşamada belirlenir.

4) Veri Dönüştürme (Data Transformation): Bu aşamada veriler veri madenciliği algoritmalarının uygulanabilmesi için uygun bir forma dönüştürülür. Veri dönüştürme işlemi veri düzeltme, birleştirme, genelleştirme ve normalleştirme gibi işlemlerin bir ya da birkaçını içerebilir.

5) Veri Madenciliği (Data Mining): Bu aşamada, anlamlı örüntüler elde edebilmek için veri üzerinde veri madenciliği algoritmaları uygulanır. Sınıflandırma, kümeleme algoritmaları gibi veri madenciliği algoritmaları kullanılarak yararlı bilgi keşfedilmesi sağlanır.

6) Örüntü Değerlendirme (Pattern Evaluation): Elde edilmiş olan bilginin basitlik, geçerlilik, yararlılık ve yenilik gibi bazı ölçüm değerlerine göre değerlendirildiği aşamadır.

7) Bilgi Sunumu (Knowledge Presentation): Bu aşamada, çeşitli görselleştirme ve bilgi sunum araçları kullanılarak elde edilmiş olan bilginin kullanıcıya sunumu gerçekleştirilir.

## 2.2. Veri Madenciliğinin Uygulama Alanları

Veri madenciliğinin birçok uygulama alanı vardır. Bu uygulama alanlarının başlıcaları aşağıda sıralanmıştır:

**Pazarlama:** Müşterilerin satın alma örüntüleri, demografik bilgileri, kampanya ürünleri belirleme, mevcut müşterileri kaybetmeden yeni müşteriler kazanma, pazar sepeti analizi (Market Basket Analysis), müşteri ilişkileri yönetimi (CRM: Customer Relations Management) ve satış tahmini alanları en yaygın veri madenciliği uygulama alanlarıdır.

**Banka ve Sigortacılık:** Farklı finansal göstergeler arasında korelasyon tespiti, kredi kartı dolandırıcılıklarının tespiti, kredi taleplerinin değerlendirilmesi, kredi kartı harcamalarına göre müşteri profili belirlenmesinde, sigorta dolandırıcılıklarının tespitinde, yeni poliçe talep edecek müşterilerin tahmininde yoğun olarak kullanılmaktadır.

**Biyoloji, Tıp ve Genetik:** Bitki türleri ıslahı, gen haritasının analizi ve genetik hastalıkların tespiti, kanserli hücrelerin tespiti, yeni virüs türlerinin keşfi ve sınıflandırılması, fizyolojik parametrelerin analizi ve değerlendirilmesinde kullanılmaktadır.

**Kimya:** Yeni kimyasal moleküllerin keşfi ve sınıflandırılması, yeni ilaç türlerinin keşfinden kullanılmaktadır.

**Yüzey Analizi ve Coğrafi Bilgi Sistemleri:** Bölgelerin coğrafi özelliklerine göre sınıflandırılması, kentlerde yerleşim yerleri belirleme, kentlerde suç oranı,

zenginlik-yoksulluk, köken belirleme, kentlere yerleştirilecek posta kutusu, otomatik para makineleri, otobüs durakları gibi hizmetlerin konumlarının tespitinde kullanılmaktadır.

**Görüntü Tanıma ve Robot Görüş Sistemleri:** Çeşitli algılayıcılar aracılığı ile tespit edilen görüntülerden yola çıkarak engel tanıma, yol tanıma, yüz tanıma, parmak izi tanıma gibi tekniklerde kullanılmaktadır.

**Uzay Bilimleri ve Teknolojisi:** Gezegen yüzey şekillerinin ve gezegen yerleşimleri, yeni galaksiler keşfi, yıldızların konumlarına göre gruplandırılmasında kullanılmaktadır.

**Meteoroloji ve Atmosfer Bilimleri:** Bölgesel iklim, yağış haritaları oluşturma, hava tahminleri, ozon tabakası deliklerinin, tespiti, çeşitli okyanus hareketlerinin belirlenmesinde kullanılmaktadır.

**Sosyal Bilimler ve Davranış Bilimleri:** Kamuoyu yoklamaları inceleme, genel eğilim belirleme, seçim öngörülerini oluşturmada kullanılmaktadır.

**Metin Madenciliği:** Çok büyük ve anlamsız metin yığınları arasından anlamlı ilişkiler elde etmekte kullanılmaktadır.

**İş ve Elektronik Ticaret Verileri:** Geri ofis, ön ofis ve ağ uygulamaları iş süreçleri sırasında geniş çaplarda veri üretirler. Bu veriyi karar verme mekanizmalarında efektif olarak kullanmak, ilgili ticari kuruluşun temel yapı taşlarından olmalıdır.

**Bilimsel, Mühendislik ve Sağlık Bakım Verileri:** Günümüzde bilimsel veriler, iş sahası verilerinden daha da karmaşık hale gelmişlerdir. Buna ek olarak, bilim adamları ve mühendisler uygulama sahası bilgilerini kullanarak benzetim ve sistem kullanımının artırılması hedefindedirler.

**Web Verileri:** İnternet ve web üzerindeki veriler hem hacim hem de karmaşıklık olarak hızla artmaktadır. Sadece düz metin ve resimden başka akan (streaming) ve nümerik veriler de web verileri arasında yer almaktadır [Han and Kamber, 2005].

## 2.3. Veri Madenciliği Model ve Teknikleri

Veri madenciliği alanında kullanılan birçok model vardır. Bu modellerden bir tanesi Han'a göre iki kategori altında toplanan tanımlayıcı (descriptive) ve öngörüşel (predictive) modellerdir. Tanımlayıcı veri madenciliği, veritabanı içindeki verinin genel özelliklerinin ortaya çıkarır. Öngörüşel veri madenciliği, gelecekle ilgili tahminlerde bulunmak için geçerli olan veri üzerinde sonuçlar çıkarmaya çalışır. Han'ın modelini kullananlar bile hangi kategorinin hangi modelin altında olduğunu kararını verememişlerdir. Bu kategoriler tanımlama ve ayırlama (characterization and discrimination), birliktelik analizi (association rules), sınıflandırma ve öngörü (classification and prediction), kümeleme analizi (cluster analysis), sıra dışılık analizi (outlier analysis) ve gelişimsel analizdir (evolution analysis). Önerilen bir diğeri veri madenciliği modellerinden biri de Berry ve Linoff tarafından ileriye sürülmüştür. Buna göre veri madenciliği yönlendirilmiş (directed) ve yönlendirilmemiş (undirected) olmak üzere iki kategoriye ayrılmıştır. Yönlendirilmiş veri madenciliği, gelir gibi belirli hedef alanlarını açıklamaya, sınıflamaya ve belirli hedef değişkenlerinin değerlerini bulmaya çalışır. Yönlendirilmemiş veri madenciliği, belirli hedef alanları veya önceden tanımlanmış sınıf topluluklarını kullanmadan kayıtlar arasındaki benzerlikleri veya örüntüleri bulmaya çalışır. Veri madenciliği tekniklerinden olan sınıflandırma, kestirim ve tahmin yönlendirilmiş veri madenciliğine örnektir. Benzer gruplama ve kümeleme teknikleri yönlendirilmemiş veri madenciliğine örnektir. Tanımlama ve belgeleme tekniği ise yönlendirilmiş veya yönlendirilmemiş veri madenciliğinden ikisinden birine girmektedir [Berry and Linoff, 2004].

### 2.3.1. Sınıflandırma (Classification)

Sınıflandırma veri madenciliğinde en yaygın olarak kullanılan tekniklerden biridir. İnsan doğası gereği dünyayı anlamak ve iletişim kurmak için çevresindeki hemen her şeyi sınıflandırmaktadır. İnsanlar, yasayan varlıkları filum, tür ve cins; maddeleri elementlere; köpekleri cinslere; insanları ırklara ayırmışlardır. Sınıflandırma yeni sunulan nesnenin özelliklerini incelemeyi ve bu nesneyi önceden tanımlanmış sınıflar kümesine atamayı içerir. Sınıflandırılacak nesnelere veritabanı tablosundaki veya bir dosyadaki kayıtlardan alınmaktadır. Sınıflandırma, sınıf koduyla yeni kolonların eklenmesinden oluşur [Berry and Linoff, 2004].

Sınıflandırma iki adımdan oluşmaktadır. Birinci adımda sınıflar iyi bir şekilde tanımlanır ve tahmin için kullanılacak bir model üretilir. İkinci adımda bu model sınıflandırılmamış veriye uygulanarak sınıflar tahmin edilir. Tahmin edilen değerler sayısal ya da kategorik değerler olabilirler. Ama sınıflandırma işlemi gerçekleştirilirken üzerinde işlem yapılacak olan eğitim verisi değerleri sayısal değerler olmalıdır [Berry and Linoff, 2004].

Sınıflandırma işlemine; kredi başvurularının düşük, orta ve yüksek riskli olarak sınıflandırılması, web sayfasında gösterilecek içeriklerin belirlenmesi, hangi telefon numaralarının faks makinelerine karşılık geldiğinin belirlenmesi, dolandırıcı sigorta vaatlerinin belirlenmesi, kredi kartı sahtekârlıklarının tespit edilmesi ve endüstri kodlarının tayini ve serbest metin is tanımına göre is atamaları örnek olarak verilebilir. Karar ağaçları, yapay sinir ağları (artificial neural networks), k-en yakın komşu (k-nearest neighbour), genetik algoritmalar (genetic algorithms), katı küme yaklaşımı (rough set approach), bulanık küme yaklaşımı (fuzzy set approach) teknikleri başlıca sınıflama teknikleridir.

### **2.3.2. Kestirim (Estimation)**

Sınıflandırma ile kestirim birbirine çok benzemektedir. Sınıflandırma, evet veya hayır; kızamık, kızamıkçık veya suçüçeğı gibi ayrık sonuçlarla uğraşmaktadır. Kestirim ise sürekli değerlendirilmiş sonuçlarla uğraşır. Kestirim, bilinmeyen sürekli değişkenler (gelir, yükseklik, kredi kartı dengesi) için bir değer bulur. Sınıflandırma ile kestirim arasındaki belirgin fark; sınıflandırma sınıfsal değerlerle ilgilenirken, kestirim rakamsal değerlerle ilgilenir. Regresyon modelleri ve nöron ağları kestirim işlemine iyi uyarlar [Berry and Linoff, 2004]. Uygulamada, kestirim sınıflama işi için kullanılır. Örneğin; kayak ayakkabısı üreticisi bir firmaya reklam alanı satmak isteyen bir kredi kartı şirketi, tüm kart sahiplerini kayakçılar ve kayakçı olmayanlar diye iki sınıfa ayıran bir sınıflama modeli oluşturabilir. Başka bir yaklaşım ise her kart sahibine bir kaymaya eğilim skoru atayan bir model oluşturmaktır. Bu değer 0' dan 1' e kadar bir değer olabilir ve bu da kart sahibinin bir kayakçı olma olasılığının tahminidir. Sınıflama görevi bir eşik skorun belirlenmesidir. Eşik değerinin üstünde veya eşik değerine eşit bir skora sahip olan herkes kayakçı olarak sınıflandırılır. Eşik değerinin altındakiler kayakçı olmayanlar olarak sınıflandırılır [Berry and Linoff, 2004].

Kestirim yaklaşımının, bireysel kayıtların tahmine göre sıralanabildiğinden büyük bir avantajı vardır. Bunun önemini görmek için kayak ayakkabısı üretici firmanın reklam için 500.000 adetlik bir bütçe ayırdığını hayal edelim. Eğer sınıflandırma yaklaşımı kullanırsak ve kredi kartı müşterilerinden 1,5 milyon kayakçı tanımlandığını varsayarsak, algoritmamız havuzdan (1,5 milyon kişiden) rastlantısal olarak seçilen 500.000 kişinin kredi kartı hesap zarfına bu reklamı koyacaktır. Kestirime göre ise her kart sahibini bir kaymaya eğilim skoru olacak ve reklâm kayakçı olma olasılığı en fazla olan 500.000 adaya gönderilecektir [Berry and Linoff, 2004].

Kestirim işlemine; bir ailedeki çocuk sayısının tahmini, bir ailenin toplam gelirinin tahmini, bir müşterinin yaşam kalitesinin tahmini, bir kişinin dengeli aktarım talebine cevap verip vermeyeceğinin olasılık tahmini örnek olarak verilebilir.

### **2.3.3. Tahmin (Prediction)**

Tahmin, kestirim ve sınıflandırma ile aynıdır. Tahminde kayıtların sınıflanması, gelecek bir davranışı veya gelecek bir değeri tahmin etmeye göre yapılır ve bu yönüyle kestirimden ve sınıflandırmadan ayrılır. Tahmin tekniğinde, sınıflandırmanın doğruluğunu test etmenin tek yolu bekleyip görmektir [Berry and Linoff, 2004].

Sınıflandırma ve kestirim için kullanılan tüm teknikler, eğitim örnekleri kullanarak tahmine uygun hale getirilebilir ve tahminde kullanılabilir. Tarihsel veri, günümüzde gözlenen davranışı açıklayan model oluşturmak için kullanılır. Tarihsel veriden oluşturulan model, günümüz verisine uygulandığında gelecekte gözlenecek davranışın tahmini meydana getirilmiş olur. Bu durumda, mevcut durum; daha önce olmuş olayları, yeni durum ise gelecekte olacak olayları tanımlar [Berry and Linoff, 2004].

Tahmin metoduna; kredi kartı müşterisi hesap transfer teklifini kabul ettiğinde transfer edilecek hesabın büyüklüğünün tahmini, 6 ay içinde hangi müşterinin ayrılacağı tahmini, hangi telefon abonesinin üç hatlı arama, sesli mesaj gibi ücret ilavesi gerektiren servisler sipariş edeceğinin tahmini gibi örnekler verilebilir [Berry and Linoff, 2004].

### 2.3.4. Benzer Gruplama (Affinity Grouping) /Birliktelik Analizi

Benzer gruplamanın görevi; hangi nesnelere birlikte olduklarını ve birbirlerini nasıl etkilediklerini bulmaktır. Bir süpermarkette alışveriş arabasında hangi malzemelerin beraber alındığını belirlemek benzer gruplamaya örnek olarak verilebilir. Market zincirleri benzer gruplamayı; bir dükkânın rafını veya bir katalogdaki nesnelere düzenlemek için kullanırlar. Müşteriler tarafından daha çok birlikte alınan ürünler aynı yere yerleştirilmektedir ve bu da müşterileri bir ürünü aldığı anda onunla ilişkili diğer ürünü de almak için teşvik etmektedir [Berry and Linoff, 2004].

Benzer gruplama, ayrıca çapraz-satış fırsatlarını tanımlamak ve ürünleri ve servisleri gruplayarak göze hoş gelen çekici paketler oluşturmak için kullanılabilir. Cep partner, cep öğrenci gibi servis paketleri örnek olarak verilebilir [Berry and Linoff, 2004].

Benzer gruplama, veriden kurallar üretmek için basit yaklaşımlardan biridir. Eğer iki nesne yeterince birlikte alınıyorsa bunlar için birliktelik kuralları (association rules) oluşturabilir. Birliktelik kuralına örnek verecek olursak; kedi maması alan kişiler P1 olasılığı ile kedi atık kovası da alırlar, kedi atık kovası alan kişiler P2 olasılığı ile kedi maması da alırlar.

### 2.3.5. Kümeleme (Clustering)

Kümeleme, heterojen olan büyük bir grubu daha homojen alt gruplara veya kümelere ayırma işlemidir. Kümeleme işleminde amaç; küme içi benzerliği maksimum ve kümeler arası benzerliği ise minimum yapmaktır. Küme içindeki benzerliğin mükemmelliği ve kümeler arasındaki farklılığın mükemmelliği kümenin daha iyi ve daha açık olmasını sağlar. Kümeleme, denetimsiz sınıflandırma yöntemidir. Kümelemeyi, sınıflandırmadan ayıran en önemli özellik kümelemenin önceden tanımlanmış sınıflara bağlı olmamasıdır. Denetimli sınıflandırma yönteminde ise yeni, sınıflandırılmamış veri önceden oluşturulan var olan sınıflardan uygun olanına yerleştirilmektedir. Kümeleme de önceden tanımlanmış bir sınıf yoktur. Kayıtlar kendi aralarındaki benzerliklere göre gruplandırılırlar. Kümeleme ile ortaya çıkan kümelerin ne anlam taşıdığını belirlemek kullanıcıya bağlıdır.

Kümeleme genellikle veri madenciliği ve modelleme tekniklerinde başlangıç olarak kullanılır. Market bölümlenmesinde kümeleme ilk adım olabilir. “Hangi tür

promosyonlara müşteriler en iyi cevabı verir ya da hangi tür promosyonlar müşterileri alışverişe yönlendirir?” sorusuyla uğraşmak yerine, önce müşteriler benzer alışveriş alışkanlıklarına göre alt kümelere bölünür ve ondan sonra hangi tür promosyonların her bir küme için en iyi sonuç sağladığı sorusu sorulur [Berry and Linoff, 2004].

### **2.3.6. Tanımlama ve profil oluşturma (Description and Profiling)**

Veri madenciliğinin amacı; bazen verinin üretmiş olduğu insanlar, ürünler ve işlemler hakkındaki bilgimizi ve anlayışımızı arttırmak için karmaşık bir veri tabanında neyin olup bittiğini tanımlamaktır. Yeterince iyi tanımlanmış bir davranış kendisi için iyi bir açıklama sunar. İyi bir tanımlama, bir açıklamayı nereden aramaya başlayacağımızı gösterir. Amerikan politikasındaki ünlü cinsiyet ayrılığı “kadınlar demokratları erkeklerden daha çok destekliyorlar” olayı basit bir tanımlamamın nasıl ilgileri üstüne topladığını göstermektedir. Bu tanımlamadan dolayı gazeteciler, sosyologlar, ekonomistler ve politika bilimcileri bu konu hakkından daha ileri çalışmalar yapmışlardır.

### 3. Birliktelik Analizi ve Apriori Algoritması

Veri madenciliğinde kullanılan ilk tekniklerden birisi de birliktelik kurallarıdır [Agrawal and Srikant, 1994] Birliktelik kuralı, geçmiş verilerin analiz edilerek bu veriler içindeki birliktelik davranışlarının tespiti ile geleceğe yönelik çalışmalar yapılmasını destekleyen bir yaklaşımdır. Birliktelik kuralı madenciliğinin uygulamasına Pazar sepeti analizi örnek verilebilir [Frawley et al., 1991]. Birliktelik kuralındaki amaç; alışveriş esnasında müşterilerin satın aldıkları ürünler arasındaki birliktelik ilişkisini bulmak, bu ilişki verisi doğrultusunda müşterilerin satın alma alışkanlıklarını tespit etmektir. Satıcılar, keşfedilen bu birliktelik bağıntıları ve alışkanlıklar sayesinde etkili ve kazançlı pazarlama ve satış imkanına sahip olmaktadır. Örneğin, bir marketten müşterilerin süt ve peynir satın almalarının % 70’inde bu ürünler ile birlikte yoğurt da satın alınmıştır. Bu tür birliktelik örüntüsünün tespit edilebilmesi için, örüntü içinde yer alan ürünlerin birden çok satın alma hareketinde birlikte yer alması gerekir. Milyonlarca veri üzerinde veri madenciliği teknikleri uygulandığında, birliktelik sorgusu için kullanılan algoritmalar hızlı olmalıdır [Agrawal and Srikant,1995]

Bu çalışmada, veri madenciliği tekniklerinden birliktelik kuralı yöntemlerinden en çok bilinen Apriori algoritmasının örnek bir firmanın sevkiyat verileri üzerinde uygulanması için web tabanlı bir uygulama yazılımı geliştirilmiştir . Geliştirilen uygulama yazılımı ile örnek firmanın sevkiyatları incelenerek aynı rota üzerinde bulunan bayiler bulunmuştur.

#### 3.1. Birliktelik Kuralı

Birliktelik kuralının matematiksel modeli Agrawal, Imielinski ve Swami tarafından yılında sunulmuştur [Agrawal and Srikant, 1994]. Bu modelde,  $I = \{i_1, i_2, \dots, i_m\}$  kümesine “ürünler” adı verilmektedir.  $D$ , veri bütünlüğündeki tüm hareketleri,  $T$  ise ürünlerin her bir hareketini simgeler.  $TID$  ise, her harekete ait olan tek belirteçtir.

Birliktelik kuralı şu şekilde tanımlanabilir;

$$A_1, A_2, \dots, A_m \Rightarrow B_1, B_2, \dots, B_n$$

Bu ifadede yer alan,  $A_i$  ve  $B_j$ , yapılan iş veya nesnelere. Bu kural, genellikle “ $A_1, A_2, \dots, A_m$ ” iş veya nesnelere meydana geldiğinde, sık olarak “ $B_1, B_2, \dots, B_n$ ” iş veya nesnelere aynı olay veya hareket içinde yer aldığını belirtir [Zhu, 1998].

Birliktelik kuralı, kullanıcı tarafından minimum değeri ( $min_{des}$ ) belirlenmiş destek ve güvenilirlik eşik değerlerini sağlayacak biçimde üretilir. Bir ürün kümesindeki destek,  $D$  ile ifade edilen tüm hareketler içinde ilgili ürün kümesini içeren hareketlerin yüzdesidir.  $A$  ve  $B$  ürün kümelerinin, birliktelik kuralı “ $A \Rightarrow B$ ” olarak gösterilirse, destek aşağıdaki gibi tanımlanır.

$$\text{destek}(A \Rightarrow B) = (\text{A ve B' nin bulunduğu satır sayısı}) / (\text{toplam satır sayısı})$$

$A \Rightarrow B$  birliktelik kuralının güven değeri ise,  $A'$  yi içeren hareketlerin  $B'$  yi de içermeye yüzdesidir. Örneğin, bir kural % 85 güvenilirliğe sahip ise,  $A'$  yi içeren ürün kümelerinin % 85'i  $B'$  yi de içermektedir. İşe bağlı veri satırları verilmiş ise, ( $A \Rightarrow B$ ) güveni aşağıdaki gibi tanımlanır.

$$\text{güven}(A \Rightarrow B) = (\text{A ve B' nin bulunduğu satır sayısı}) / (\text{A' nın bulunduğu satır sayısı})$$

Güven değerinin % 100 olması durumunda, kural bütün veri analizlerinde doğrudur ve bu kurallara “kesin” denir. Birliktelik kuralına ilişkin olarak geliştirilen bazı algoritmalar şunlardır; AIS [Agrawal and Srikant, 1994], SETM [33], Apriori [Agrawal and Srikant, 1995], RARM - Rapid Association Rule Mining [34], CHARM [35]. Bu algoritmalar içerisinde, ilk olanı AIS, en bilineni ise Apriori algoritmasıdır [Agrawal and Srikant, 1995].

### 3.2. Apriori Algoritmasının Yapısı

Apriori Algoritmasının ismi, bilgileri bir önceki adımdan aldığı için “prior” anlamında Apriori'dir [Agrawal and Srikant, 1994]. Bu algoritma temelinde iteratif (tekrarlayan) bir niteliğe sahiptir ve hareket bilgileri içeren veritabanlarında sık geçen öğe kümelerinin keşfedilmesinde kullanılır. Apriori Algoritmasına özüne göre, eğer  $k$ -öge kümesi ( $k$  adet elemana sahip öğe kümesi)  $min_{des}$  ölçütünü sağlıyorsa, bu kümenin alt kümeleri de  $min_{des}$  ölçütünü sağlar. Birliktelik kuralı madenciliği, tüm sık geçen öğelerin bulunması ve sık geçen bu öğelerden güçlü birliktelik kurallarının üretilmesi olmak üzere iki aşamalıdır. Birliktelik kuralının ilk aşaması için kullanılan Apriori Algoritması, sık geçen öğeler madenciliğinde kullanılan en popüler ve klasik

algoritmadır. Bu algoritmada özellikler ve veri, Boolean ilişki kuralları ile değerlendirilir [Gao, 2004].

k-öge (k adet elemana sahip öge kümesi) kümesi  $c$  ile ifade edilirse, öğeleri (ürünler)  $c[1], c[2], c[3], \dots, c[k]$  şeklinde gösterilir ve  $c[1] < c[2] < c[3] < \dots < c[k]$  olacak şekilde küçükten büyüğe doğru sıralıdır [Agrawal and Srikant, 1994]. Her öge kümesine destek ölçütünü tutmak üzere bir sayaç değişkeni eklenmiştir ve sayaç değişkeni öge kümesi ilk kez oluşturulduğunda sıfırlanır. Geniş (sık geçen) öge kümeleri  $L$  karakteri ile, aday öge kümeleri ise  $C$  karakteri ile gösterilir [Sever ve Oğuz, 2002].

```

1)  $L_1 = \{\text{sık geçen 1-öge kümesi}\};$ 
2) for ( $k=2; L_{k-1} \neq \emptyset; k++$ ) do begin
3)    $C_k = \text{apriori-gen}(L_{k-1});$  // Yeni adaylar
4)   forall transactions-hareketler  $t \in D$  do begin
5)      $C_t = \text{subset}(C_k, t);$  // Adaylar  $t$  içindedir
6)     forall candidates – adaylar  $c \in C_t$  do
7)        $c.\text{count}++;$ 
8)   end
9)    $L_k = \{c \in C_k \mid c.\text{count} \geq \text{minsup}\}$ 
10) end
11) Answer =  $\cup_k L_k;$ 

```

Şekil 3.1. Klasik Apriori Algoritması Özet Kodu [Özçakır ve Çamurcu, 2007]

Apriori algoritmasının klasik özet kodu Şekil 3.1’ de [Agrawal and Srikant, 1994] görülmektedir. Bu şekilde yer alan apriori-gen fonksiyonu (Şekil 3.2) [Agrawal and Srikant, 1994],  $(k-1)$  adet öğeye sahip  $L_{k-1}$  sık geçen öge kümesini kullanarak  $k$  adet öğeye sahip aday kümeleri oluşturur. Bu fonksiyon ile, ilk önce,  $L_{k-1}$  sık geçen öge kümesine kendisi ile birleştirme (join) işlemi uygulanır. Birleştirme işleminde  $L_{k-1}$  sık geçen öge kümesinin her satırında yer alan son öge haricinde diğer öğelerin çapraz olarak benzerliği aranır ve son öge haricinde diğer öğelerle yakalanan benzerliklerden yeni aday öge kümeleri oluşturulur. Oluşan kümeler budama (prune) adımı ile budanarak fonksiyondan dönlür.

```

insert into  $C_k$ 
select  $p.items_1, p.items_2, \dots, p.items_{k-1}, q.item_{k-1}$ 
from  $L_{k-1} p, L_{k-1} q$ 
where  $p.item_1=q.item_1, \dots, p.item_{k-2}=q.item_{k-2},$ 
 $p.item_{k-1} < q.item_{k-1};$ 

forall itemsets  $c \in C_k$  do
  forall  $(k-1)$ -subsets  $s$  of  $c$  do
    if  $(s \notin L_{k-1})$  then
      delete  $c$  from  $C_k$ 

```

Şekil 3.2. Apriori-Gen Fonksiyonu [Özçakır ve Çamurcu, 2007]

Budama işleminde;  $c$  aday kümesinin  $(k-1)$  öğeye sahip alt kümelerinden  $L_{k-1}$  sık geçen öğe kümesinde yer almayan tüm alt kümeler silinir. Farklı bir ifade ile budama,  $C_k$  aday öğe kümesindeki öğelerin alt kümelerinin  $L_{k-1}$  sık geçen öğe kümesindeki varlığı kontrol edilir, bir öğenin alt kümelerinden biri,  $L_{k-1}$  sık geçen öğe kümesinde yer almıyorsa ilgili öğe değerlendirme dışı kalır ve  $C_k$  aday öğe kümesinden silinir [Agrawal and Srikant, 1994].

Apriori algoritması özet kodu incelendiğinde sık geçen öğe kümelerini bulmak için bir çok kez veritabanının tarandığı görülmektedir. İlk aşamadan önce, veri madenciliği uygulanacak veri topluluğunun taranarak öğelerin kaç adet hareket kaydı içinde yer aldığı tespit edildiği (her öğe için tespit edilen bu değere destek sayacı adı verilir) ve destek sayacı minimum destek değerine eşit veya büyük olan öğelerin  $L_1$  sık geçen 1-öge kümesi olarak belirlendiği varsayılarak işleme başlanır. Kod içinde kurulan döngü yapısı ile ilk aşamada  $L_1$  sık geçen öğe kümesinin öğelerinin ikili kombinasyonuna benzer bir şekilde  $(L_1 \infty L_1)$  yeni bir küme oluşur, bu işleme birleştirme (join) adı verilir. Bu işlem ile oluşan kümelere de aday öğe kümeler adı verilir ve  $C$  harfi ile simgelenir. Oluşan bu aday öğe kümesinin her elemanı iki adet öğeden oluştuğu için  $C_2$  ifadesi ile isimlendirilir. Bu aday küme apriori-gen işlevi ile budama işlemine tabi tutulur ve  $C_2$  kümesinin elemanlarına ait alt kümelerinin  $L_1$  öğe kümesinde olup olmadığına bakılır, alt kümelerden  $L_1$  içinde yer almayan küme elemanları  $C_2$  aday kümesinden silinir. Apriori algoritması uygulanan veri topluluğu tekrar taranarak budama işleminden geçen  $C_2$  aday kümesi elemanlarının kaç adet hareket kaydı içinden geçtiği (destek sayacı) bulunur. Bulunan destek sayacı bilgileri doğrultusunda  $C_2$  aday kümesi elemanlarının destek sayacı minimum destek değerine eşit veya büyük destek değerine sahip olan elemanları  $L_2$  sık geçen öğe kümesini

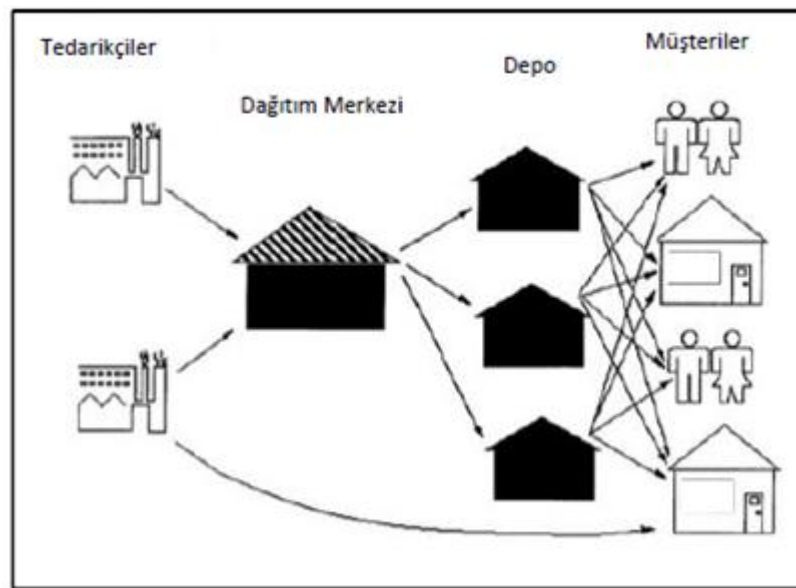
oluřturur. Döngü bir sonraki ařamada  $L_2$  kümesi öğelerinin üçlü kombinasyonu ile yeni bir aday öge kümesi oluřturur ve bu küme  $C_3$  ifadesi ile simgelenir. İlk ařamada olduđu gibi bu kümede budama iřleminden geđer ve budama iřleminden sonra minimum destek seviyesinin üstünde kalan elemanları ile  $L_3$  sık geđen öge kümesi oluřturulur. Döngü her dönuřünde öge sayısını artırarak devam eder. Bu süreç yeni bir sık geđen öge kümesi bulunamayana kadar sürer.

## 4. ARAÇ ROTALAMA

### 4.1. Lojistik

#### 4.1.1. Lojistik Nedir?

Günümüz küresel piyasasında yoğun rekabet, kısa yaşam eğrisine sahip ürünler ve müşterilerin artan beklentileri, üreticileri dağıtım sistemlerine önem vermeye ve yatırım yapmaya zorlamıştır. Bu durum, iletişimdeki ve ulaşım teknolojilerindeki değişimle birlikte, örneğin mobil iletişim ve gıda dağıtım gibi, lojistik yönetiminin sürekli gelişimine neden olmuştur [Simchi-Levi and Bramel, 1997]



Şekil 4.1. Ağ [Simchi-Levi and Bramel, 1997]

Lojistik sistemlerinde mallar bir veya daha fazla fabrikada üretilmekte, depolama için depolara dağıtılmakta ve daha sonra perakendecilere veya müşterilere dağıtılmaktadır. Sonuç olarak, maliyetleri azaltmak ve servis seviyesini iyileştirmek için, bu dağıtım ağındaki farklı seviyelerin etkileşimlerini dikkate almak gerekir. Bu ağ, Şekil 4.1’de görüldüğü gibi tedarikçiler, üretim merkezleri, depolar, dağıtım merkezleri ve perakendeci mağazalarından oluşmaktadır [Simchi-Levi and Bramel, 1997].

Hesket ve diğerlerine göre (1970) “ Lojistik, zaman ve yer faydası yaratmada, arz ve talebin koordinasyonu ve hareketini kolaylaştıran tüm faaliyetlerin yönetimidir”.

Wikipedia' da (2006). “ Lojistik, ürün, enerji, bilgi ve diğer kaynakların akışını yönetme ve kontrol etme sanat ve bilimidir” şeklinde tanımlanmıştır.

CSCMP tarafından (2006). “ Lojistik yönetimi, müşteri isteklerini karşılamak için, başlangıç ve tüketim noktası arasında mal, hizmet ve ilgili bilgilerin etkin, etkili gönderimi ve tersine akışını ve depolanmasını planlamak, uygulamak ve kontrol etmektir” olarak tanımlanmıştır.

Lojistik ve Ulaşım Enstitüsü'ne göre (UK, 2005) “Lojistik, kaynağın doğru zamanda, doğru yerde, doğru maliyette, doğru kalitede konumlandırılmasıdır”.

Birçok sektöre uygulanan uygun modern bir tanım, “lojistik, müşteriye kabul edilebilir hizmet sunmakla birlikte, maliyet etkin bir yolla, arz kaynağından üretim yeri aracılığıyla tüketim noktasına, ürünlerin etkin bir şekilde transfer edilmesiyle ilgilidir” şeklinde yapılmaktadır [Rushton et al., 2006].

Lojistik Yönetimi Kurulu'na göre lojistik yönetimi, “müşteri isteklerine uygunluk sağlamak amacıyla, merkez noktadan tüketim noktasına mal, hizmet ve ilgili bilgilerin etkin ve etkili akışı ve depolanışının planlanması, uygulanması ve kontrol edilmesi sürecidir” [Simchi-Levi and Bramel, 1997]. Bu tanımlamaya göre birinci olarak, lojistik yönetimi, sistem etkililiği üzerinde etkisi olan ve müşteri isteklerine uygun ürün üretmede rol oynayan, tedarikçi ve üretim merkezlerinden depo ve dağıtım merkezleri aracılığıyla perakendeci ve mağazalara kadar her merkezi dikkate almaktadır. İkinci olarak, lojistik yönetiminin amacı tüm sistemde etkin bir maliyete sahip olmaktır; amaç hammadde, yarı mamul ve mamullerin dağıtımını ve ulaşımından stoklanmasına kadar tüm sistemin maliyetlerini minimize etmektir. Dolayısıyla odak noktası sadece ulaşım maliyetlerini minimize etmek ve stokları azaltmak değil, daha çok lojistik yönetimine bir sistem yaklaşımıdır. Son olarak, lojistik yönetimi, lojistik ağını planlamak, uygulamak ve kontrol etmek çerçevesinde geliştiğinden, stratejik seviyeden, taktik ve operasyonel seviyeye birçok firma faaliyetini içermektedir [Simchi-Levi and Bramel, 1997].

#### **4.1.2. Lojistik Yönetiminin Önemi?**

Bir bütün olarak lojistik sistemi ürün sağlanması, dağıtımını, depolanması, stok kontrolü, bilişim işlemleri ve diğer faaliyetleri içeren, bir seri faaliyetin bulunduğu bir sistemdir. Bu faaliyetlerin en azından bir kısmı her organizasyonda yer almaktadır. Christopher (1986) “Lojistik her zaman merkezde olan ve en önemli

ekonomik bir faaliyettir.” diyerek lojistik sisteminin önemini dile getirmiştir. Deloitte ve Touche'nin 1996 yılında Kanada'da yaptığı bir çalışmaya göre firmaların %98'i lojistik faaliyetlerini kritik veya çok önemli olarak görmektedirler(Waters, 2006).

Bu kadar öneme sahip olmasına karşın uzun zamandır lojistik gerektiği ilgiyi görememiştir. Geleneksel olarak firmalar tüm yoğunluğunu üretim yapmaya harcamışlar ve lojistik kısmını pek önemsememişlerdir. Buna karşın McKibbin (1982) ve Delaney (1986) lojistik maliyetlerinin ürün maliyetlerinin %15-20 arasında olduğunu öne sürmüşlerdir.

Bu noktada, lojistiği bir bütün olarak işletme ve ekonomi çerçevesinde düşünmek faydalı olmaktadır. Lojistik ulusal ekonomiyi etkileyen insan ve malzeme kaynaklarının geniş kullanımını sağlamada önemli bir faaliyettir.

Lojistiğin ekonomideki etkisinin boyutunu tahmin etmeye yönelik bazı araştırmalar yapılmıştır [Rushton et al., 2006]. Bu araştırmalardan biri, Birleşik Krallık'ta de çalışan nüfusun yaklaşık % 30'nun lojistikle ilgili işlerde çalışmakta olduğunu göstermektedir. Ulusal seviyede Amerika Birleşik Devletleri' nin gayri safi milli hasılası \$12 trilyon dolar iken bunun yaklaşık olarak \$2.4 trilyon dolarının lojistik için harcanmış olabileceği ve bunun da yarısının sadece dağıtım maliyetleri olduğu tahmin edilmektedir (Waters, 2006). Bu rakamlar birçok önemli maliyetleri temsil etmekte ve lojistik maliyetlerinin doğasını anlamanın ve bu maliyetleri minimumda tutma yolunu belirlemenin ne kadar önemli olduğunu göstermeye hizmet etmektedir.

Lojistikteki farklı unsurların maliyetlerinin ayrıntılı hesabı, birçok araştırmada incelenmiştir. Herbert W. Davis & Company (2005) tarafından yapılan Amerika Birleşik Devletleri'ndeki lojistik maliyetleri ile ilgili araştırmada, ulaşımın % 45 ile en önemli unsur olduğu, bunu % 23 ile stok bulundurma maliyetlerinin, % 22 ile depolama ve % 10 ile yönetimin takip ettiği ortaya konulmuştur (Rushton ve diğerleri, 2006).

Bu genel durum, A.T. Kearney tarafından üretilen, Avrupa lojistik verimlilik araştırması tarafından desteklenmiştir. Temel Avrupa Birliği ekonomilerini kapsayan bu sonuçlar, tüm maliyetlerin %41'inin ulaşım, %23'ünün stok bulundurma maliyetleri, %21'inin depolama ve %15'inin yönetim olduğunu göstermiştir. Dolayısıyla her iki çalışmada, dağıtımda ulaşım maliyeti unsuru en önemli bölümü

oluşturmuştur [Rushton et al., 2006]. Bu maliyetlerin göreceli oluşumunun bir firmadan diğerine ve bir sektörden diğerine nasıl değiştiğini görmek çok ilginçtir. Birleşik Krallık'taki bir danışmanlık firması tarafından yapılan sektör maliyet denetiminden, farklı firmalardan lojistik maliyeti örnekleri alınmıştır. Çeşitli firmalardan alınan sonuçlar arasında büyük farklılıklar vardır. Bu maliyet farklılıklarının bir temel nedeni, lojistik yapılarının bir firmadan diğerine ve bir sektörden diğerine çarpıcı biçimde farklı olabilmesindedir [Rushton et al., 2006].

### **4.1.3. Lojistik Sistemindeki Problemler**

Lojistik sistemi, dağıtım ağı konfigürasyonundan başlayan ve paketleme işlemine kadar süren çözülmesi gereken birçok farklı problemi içermektedir. Bu problemler Dağıtım Ağı Konfigürasyon Problemi, Üretim Planlaması Problemi, Stok Kontrol, Stok ve Dağıtımın Entegrasyonu, Araç Filosu Yönetimi, Araç Rotalama, Paketleme Problemi, Zaman Penceresiyle Dağıtım, Toplamalı Dağıtım Sistemleri. Tezle ilgili olduğundan Araç Filosu Yönetimi ve Araç Rotalama problemi detaylı anlatılmıştır.

#### **4.1.3.1. Araç Filosu Yönetim**

Bir depo sınırlı kapasiteli araç filosu kullanarak perakendecilere ürünlerini sunmaktadır. Bir gönderici, araçlara yük dağıtma ve araç rotalarını belirleme görevindedir. Birinci olarak gönderici, bir araç tarafından uygun olarak sunulan, yükü bir araca uyan gruplara perakendecinin nasıl katılacağına karar vermelidir. İkinci olarak gönderici, maliyeti minimize etmek için hangi sırayı kullanacağına karar vermelidir. Doğal olarak, iki maliyet fonksiyonundan biri mümkündür; birinci olarak amaç, kullanılan araç sayısını minimize etmek, ikinci olarak odak noktası, seyahat edilen toplam uzaklığı azaltmaktır. Sonrası, sınırlı kapasiteli araç filosu tarafından müşterilere hizmet verilmesi gereken tek depolu Kapasiteli Araç Rotalama Problemi örneğidir. Araçlar başlangıçta depoda yer almaktadır ve amaç minimum toplam uzunluğu olan araç rotaları bulmaktır (Simchi-Levi ve Bramel, 1997).

#### **4.1.3.2. Araç Rotalama**

Perakendecilere ürün dağıtmak için depodan ayrılan bir araç olduğu varsayalım. Perakendecilerin verdiği sipariş miktarı, dağıtımın ne kadar uzun olacağını ve hangi sürede aracın depoya dönebileceğini belirleyecektir. Dolayısıyla

aracın etkin bir rota izlemesi önemlidir. Bir depodan perakendecilere minimum uzunluktaki rotayı bulma problemi (hem zaman, hem uzaklık açısından) Gezgin Satıcı Problemi örneğidir ve araç

#### 4.1.4. Lojistik Problemlerini Modelleme

Önceki bölümde sunulan, matematiksel olarak tanımlanabilen problemler, gerekli veriler mevcut olmadığı sürece anlamsızdır. Verileri bulmak, doğrulamak ve tablo haline getirmek oldukça güçtür. Gerçekte stok tutma maliyetleri, üretim maliyetleri, ekstra araç maliyetleri ve depo kapasitelerini saptamak genellikle zordur. Bunun yanında, belirli lojistik problemi ile ilgili verileri tanımlamak, veri toplama problemine başka bir karmaşıklık getirmektedir. Veriler olsa bile, karmaşık gerçek hayat problemlerini modelleme ile ilgili başka güçlükler vardır. Analizlerde seyahat zamanlarında sapmalar, üretimde yer alan değişkenler, stok azaltma, tahmin, işgücü çizelgeleme gibi unsurlar genellikle dikkate alınmaz. Bu unsurlar, lojistik uygulamaları daha da karmaşıklarıdır (Simchi-Levi ve Bramel, 1997).

Firmalar lojistik problemlerini çözmek için çeşitli yaklaşımlar kullanmaktadır. Birinci olarak, insanlar geçmişte iyi olan şeyleri tekrar etmeye eğilimlidir. Eğer geçmiş yılın emniyet stok seviyesi yığın talepten kaçınmak için yeterli ise, benzer seviye bu yıl da kullanılabilir. Eğer geçmiş yılın dağıtım rotası başarılıysa, tüm perakendeciler zamanında teslim alırlar ve dolayısıyla değıştirmmezler. İkinci olarak, sıklıkla kullanılan “öncelik kuralı” çok etkin olabilir. Örneğin birçok lojistik yöneticisi sık sık “20/80 kuralını” kullanır, buna göre ürünlerin yaklaşık %20’si, toplam maliyetin yaklaşık %80’ini yaratmaktadır ve dolayısıyla bu kritik ürünlere yoğunlaşmak yeterlidir.

Lojistik ağı tasarımı çeşitli tecrübelerden de faydalanılarak yapılır. Bunun bir örneği, eğer firma ABD’de hizmet sunuyorsa ve sadece bir depoya ihtiyaç varsa, bu deponun Chicago’da yer alması gerekirken iki depoya ihtiyaç var ise birinin Los Angeles’ta diğzerinin Atlanta’da kurulması gerekir. Sonuç olarak bazı firmalar lojistik uzmanları sezgisine ve deneyimine başvurmayaya çalışırlar ve rakipleri için iyi olan şeyin kendileri için de iyi olması gerektiği fikrine sahiptirler (Simchi-Levi ve Bramel, 1997).

Tüm bu yaklaşımlar cazip ve anlam yaratan lojistik stratejilerle sonuçlanırken, belirli olaylar için “en iyi” stratejiye odaklanmamak yüzünden ne kadar kaybedildiği açık değildir.

Yakın dönemde, hızla artan bilgisayar çözüm süreleri ile, sadece büyük firmalar için değil birçok firma için, lojistik stratejilerini optimize etmek amacıyla sofistike karar destek sistemlerini almak ve kullanmak şart olmuştur. Bu sistemlerde veriler girilmekte, gözden geçirilmekte ve çeşitli algoritmalar yönetilerek önerilen sonuç sade bir şekilde sunulmaktadır. Sistem uygun problemi çözmektedir ve bu karar destek sistemleri, sistem genelindeki maliyetleri önemli ölçüde azaltmaktadır. Her ne kadar tam anlamda kesin olarak “optimizasyon” olarak düşünülme de, sistem kullanıcıları için faydalı bir araç olarak hizmet vermektedir. Çoğu olayda bu sistemler yöneylem araştırması, yönetim bilimleri ve bilgisayar bilimleri tarafından geliştirilen teknikleri uygulamaktadır (Simchi- Levi ve Bramel, 1997).

Yukarıdaki problemlerin çoğu, NP-zor problemler olarak nitelendirilen, zor birleşik problemlere sahiptir. Bu durum göstermektedir ki, her zaman optimal çözümü veya en iyi olası kararı bulacak bir algoritmayı yaratabilmek çok zordur. Dolayısıyla birçok durumda kesin çözüm yöntemleri kullanılarak optimal çözüme ulaşılamamaktadır ve böylece sezgisel yöntemler kullanılarak problemlere çözümler aranmaktadır (Simchi-Levi ve Bramel, 1997).

## 4.2. ARP Çeşitleri

Araç rotalama problemleri gerçek hayattaki bazı özel durumlardan kaynaklanan bazı kısıtlar nedeniyle çeşitli dallara ayrılır [<http://neo.lcc.uma.es/radiaeb/WebVRP>]. Bunlar:

**Karma Kapasiteli Araç Rotalama Problemi:** Araç rotalama probleminde yer alan dağıtım yapan araçların belirli bir kapasitesinin olması durumudur. Karma kapasiteli araç rotalama probleminde her bir aracın birbirinden farklı bir kapasitesi olabilir.

**Çoklu Depoya Sahip Araç Rotalama Problemi:** Dağıtım firmasının müşterilere hizmet vermek için birden fazla deposunun olması durumudur. Eğer müşteriler depoların etrafında kümelenmiş ise, dağıtım problemi ayrı birer ARP olarak modellenebilir. Ama müşteriler ve depoların yerleri birbirlerine karışmış ise, çoklu depoya sahip araç rotalama probleminin çözülmesi gerekmektedir. Bu

problemde araçlar depolara atanır ve her bir araç ait olduğu depodan çıkarak müşteriye hizmet verir ve yine aynı depoya geri döner.

**Bölünmüş Talebe Sahip Araç Rotalama Problemi:** Bölünmüş talebe sahip araç rotalama problemi, aynı müşteriye birden fazla aracın servis yapabilmesine olanak veren araç rotalama problemidir.

**Belirsiz Talebe Sahip Araç Rotalama Problemi:** Bu tür bir problem, talebin belirsiz olduğu araç rotalama problemidir. Dağıtım aracı müşteriye vardığı zaman o müşterinin talebinin ne olacağı belli olur.

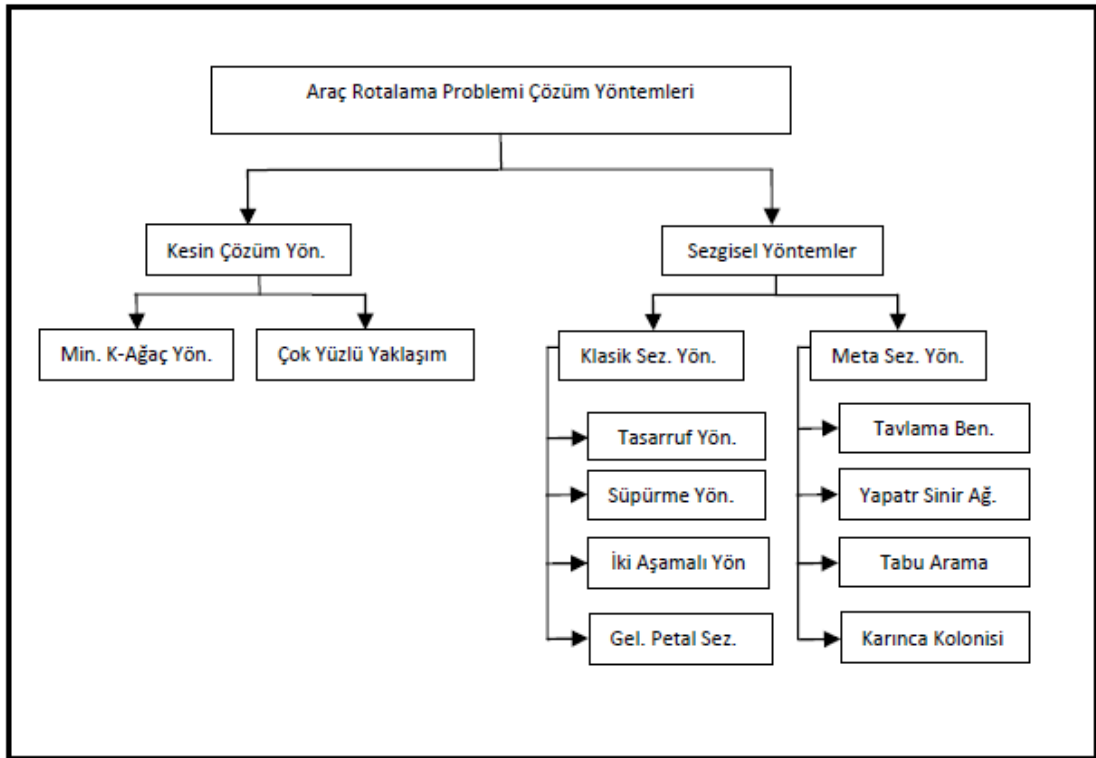
**Geri Toplaması Olan Araç Rotalama Problemi:** Geri toplaması olan araç rotalama problemi, müşterilerin depozito, ambalaj ve palet gibi, ürünlere ait bazı parçaları iade etme durumu olabilen araç rotalama problemidir. Bu durumda müşterilerden geri verilecek olan parçalar hesaba katılarak araç kapasiteleri hesaplanmalıdır.

**Zaman Pencere Araç Rotalama Problemi:** Zaman pencere Araç rotalama problemi, her bir müşteriye ait bir zaman aralığı kısıtı olan araç rotalama problemidir. Bu problemde dağıtım aracı, her bir müşteriye belirli bir zaman aralığında hizmet vermek zorundadır.

**Asimetrik Araç Rotalama Problemi:** Dağıtım aracının depodan müşteriye gidiş mesafesi ile aynı müşteriden depoya olan uzaklığın farklı olduğu araç rotalama problemine, asimetrik araç rotalama problemi denir. Bu durumda maliyet (mesafe) matrisi simetrik değildir.

### 4.3. ARP İçin Çözüm Yöntemleri

Araç rotalama problemini çözmek için araştırmacılar tarafından pek çok yöntem geliştirilmiştir. Bu çözüm yöntemleri optimal çözüme ulaşip ulaşmamasına göre kesin çözüm yöntemleri ve sezgisel yöntemler olarak ikiye ayrılır. Araç rotalama problemi için kullanılan başlıca çözüm yöntemleri Şekil 4.2'de gösterilmektedir.



Şekil 4.2. Araç Rotalama Problemi Çözüm Yöntemleri

Kesin yaklaşımların en önemli sorunu sadece küçük boyutlarda problemler için uygun olmasıdır. Mesela Fisher'ın [Fisher, 1994] önerdiği K-tree metodu 71 müşteriye kadar çözüm sunabilir. Çünkü işlem süresi ARP büyüklüğüyle birlikte üssel olarak artar [Toth and Vigo, 2002]. Sezgisel yaklaşımlarda, rota iyileştirme yöntemleri olabilecek en kabul edilebilir rotaları çıkarır. Buna örnek olarak Thompson and Psaraftis [Thompson and Psaraftis, 1993] ve Kinderwater [Kinderwater and Savelsbergh, 1997] metodları gösterilebilir. Thompson and Psaraftis [Thompson and Psaraftis, 1993] ,müşteri rotalarını içeren dairesel turlar oluşturur. Böylece permütasyon yalnızca o turdaki müşteriler arasında hesaplanır. Kinderwater [Kinderwater and Savelsbergh, 1997] ise bu dairesel turların izole olmadığı gerektiğinde müşterilerin turlar içinde değişebildiği çözümdür. Diğer bir sezgisel yöntem de iki fazlı yaklaşımlardır. Bu yaklaşımda uğranması gereken müşteriler kümelenecek bölgeler oluşturulur ardından rotalama yapılır (kümele sonra rotala). [Fisher and Jaikumar, 1981], [Taillard, 1993] Meta-sezgisel yöntemlere örnek olarak Tabu arama [Kelly and Xu, 1996] , Genetik Algoritmaları [Goldberg, 1989], [De Jong, 1975], [Holland, 1975] verebiliriz. Tabu arama [Berson et al., 1999], iyi rotaların bir havuzda saklanıp iyi çözümlerin birleştirilerek rotanın oluşturulduğu çözümdür. Genetik Algoritmalar [Goldberg, 1989], [De Jong, 1975], [Holland, 1975]

müşteriler kromozom gibi düşünülduğünde belirli özelliği sağlayan kromozom dizilimlerinin seçilerek yeni dizilerin daha sağlıklı olmasını sağlamaktadır. Genetik algoritmalar tek bir çözüm üzerinde değil bir çözüm aday uzayında çalışır. Aynı anda birden fazla çözüm adayı üretir.

#### **4.4. ARP Çözümünde Veri Madenciliği Kullanımı**

Şu ana kadar veri madenciliği teknikleri sezgisel yöntemlerin iyileştirilmesinde kullanılmıştır. Örneğin, kümeleme analizi sayesinde müşteriler benzerliklerine (coğrafi yakınlık vs.) göre gruplara ayrılabilir ve bu gruplara göre rotalama yapılabilir [Hu and Huang, 2007]. Ya da müşteri talebinin bilinmediği durumda olası talep şu ana kadar olan talep verisinden elde edilebilir [Marković et al. , 2005]. Genetik algoritmalarındaki deneme sayısını azaltmak için en iyi popülasyonlarda sık kullanılan rotalar birliktelik analizi ve sınıflandırma ile belirlenir [Santos et al., 2006]. Önerdiğimiz çözüm ise ARP problemi için sadece veri madenciliği tekniklerinin kullanıldığı bir çözümdür.

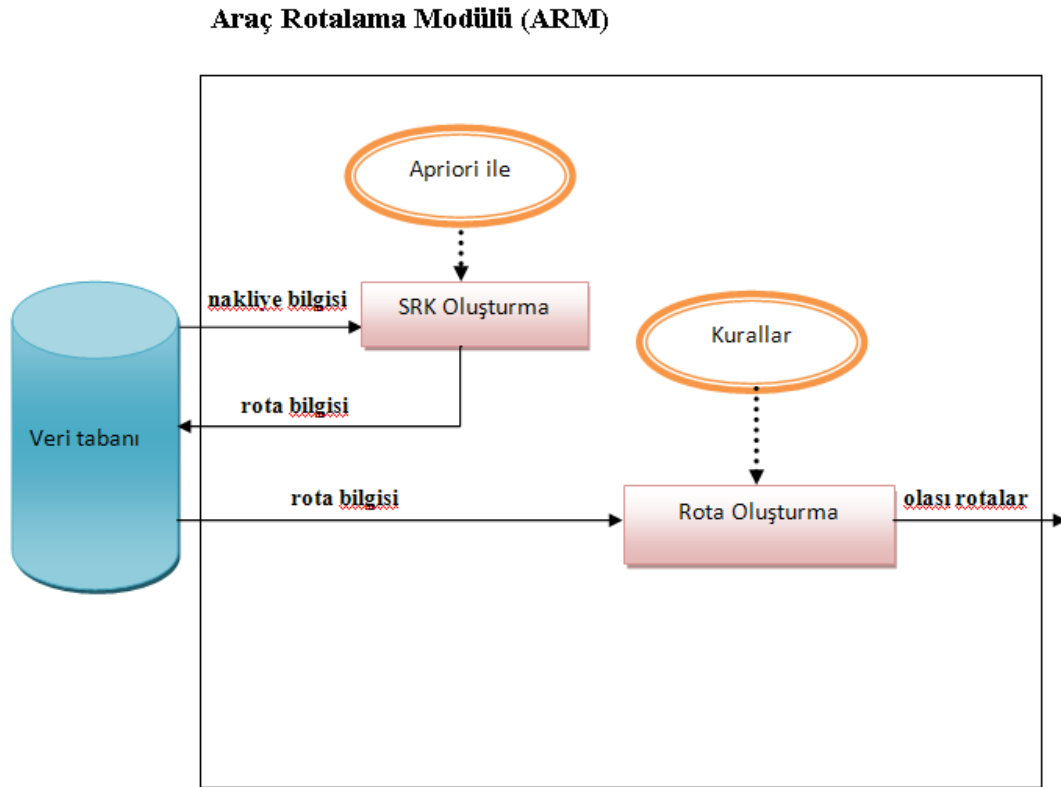
## 5. ARP' nin Veri Madenciliği Kullanarak Çözümü

İlk olarak problem doğası incelenmiştir. Elimizdeki problem stok alanındaki araçların hangi rotalar ile bayilere ulaştırılacağıdır ve problem incelendiğinde araçları aynı tır içerisinde yer alan bayilerin çok sık değişmediği gözlemlenmiştir. Şu ana kadar gönderilen araçların rotaları bundan sonraki sevkiyatlar için de kullanılması en yüksek olasılığa sahip olan rotalardır. Bu probleme uygun veri madenciliği teknikleri ve uygulamaları incelenmiştir. Problem aynı rota üzerinde giden araçların birliktelik ilişkisini bulmak olduğundan Birliktelik Analizi Tekniği çözüm olarak kullanılmıştır.

Veritabanındaki veriler var olan diğer sistemleri destek için oluşturulduğundan, veri ön işleme adımından geçirilmiştir. Bu adımda; sevkiyat sistemi veri tabanlarındaki ham verilerin bir dizi işlemten geçirilmesi ile rotalama veri tabanı oluşturulmuştur. Veritabanı Oracle 10g üzerindedir. Ardından Rotalama veri tabanı üzerinde iki aşamalı bir model kurulmuştur.(Şekil 5.1)

1-SRK(Sık kullanılan Rotalar Kümesi) Bulma

2-Rota Oluşturma (RO)



Şekil 5.1. Sistem Modeli

İlk aşama yani SRK oluşturma adımı her rotalama işleminde yenilenmesi gerekmediğinden ihtiyaca göre haftalık ya da aylık periyotlarda oluşturulabilir. İkinci aşama, tır rotalarının belirlenmesi ise park alanında bekleyen araç sayısına göre günde en az 1-2 kez yapılabilecek bir işlemdir. Bu yüzden kabul edilebilir bir zamanda çözüm üretecek şekilde tasarlanmıştır.

Sık Kullanılan Rotaların izlendiği ve sonunda oluşan araç rotalarının izlendiği bir web arayüzü tasarlanmıştır. Arayüz Oracle Apex 3.1 ile gerçekleştirilmiştir. Araç Rotalama Sistemi adı verdiğimiz uygulama Şekil 4.2'deki gibidir. Bu uygulamada 2 işlem vardır. Birincisi SRK Oluşturma' dır. Oluşturulan SRK' ların izlendiği ve düzenlendiği ekrandır. İkincisi ise var olan SRK' lara göre rotalamanın yapıldığı, izlendiği ve düzenlendiği ekrandır.



Şekil 5.2. Sistemi Giriş Sayfası

## 5.1. SRK Oluşturma

SRK bulma adımında Apriori algoritması gerçekleştirilerek sık kullanılan rotalar ortaya çıkarılmıştır. Bu algorithmada aynı rota üzerinde bulunan merkezler veritabanı taranarak bulunur ve minimum destek değerine ulaşan kayıtlar sık kullanılan rota olarak değerlendirilir. Minimum destek değeri o rotanın sık kullanılan

rota olarak işaretlenmesi için ulaşması gereken tekrarlanma sayısıdır. İlk aşama sık kullanılan rotaların ortaya çıkarıldığı, aynı zamanda verilerin de incelenip karakteristiklerinin ortaya çıkarıldığı adım olmuştur. Bu karakteristiklerin neler olduğu aşağıda açıklanmıştır.

Rotaların bulunduğu tablo çeşitli sorgularla incelendiğinde eğer aynı ilde iki bayi varsa tırların öncelikle aynı ilde bulunan bayiler ile yüklendiği gözlemlenmiştir. Ancak, İstanbul için durum biraz farklıdır. İkisi de İstanbul'da olan iki bayi aynı yükte hiç olamayabilmektedir. Bu veriler incelendiğinde İstanbul'daki bayiler için aynı yakada olma gibi bir ölçüt daha eklenmesi gerekmektedir. Bu durumda veriler için yapılan ilk analizden aşağıdaki gibi bir kural çıkartılmıştır ve bulunan kural ikinci adımdaki rota belirleme adımına bir kural olarak eklenmiştir.

**Kural 1:** Tır oluşturulurken eğer tek bir bayi ile tır'ı tamamlayamıyorsak öncelikle aynı ildeki başka bir bayinin araç isteğiyle tır'ı tamamlayabiliyor muyuz ona bakarız (İstanbul ise aynı yakada olup olmamasına da bakmamız gerekir). Yani aynı ildeki bayiler seçilerek yapılan tırların değeri maksimumdur.

Bu kural oluşturulduktan sonra; aynı ilde bulunan bayilerle ilgili ilişkiler bu veriden çıkartılır. Bu ilişkiler Kural 1 ile ikinci aşamada yönetilebilir. Oluşturulan tabloda hangi bayinin hangi bayi ile hangi sıklıkta aynı tırda gönderildiği bilgisi yer alır. Ama hangi ille hangi ilin aynı rotada bulunduğu bilgisine erişmek istendiğinden aynı il içindeki bayileri de gruplayarak hangi illerin hangi iller ile aynı rota üzerinde bulunduğu çıkartılır. Minimum destek değeri olarak ikili rotalar için 10, üçlü rotalar için 5, dörtlü rotalar için ise 3 alınır. Ancak tekrarlanma sıklığı da bir değer olarak veritabanına kaydedilir çünkü olası il taramaya şu ana kadar en sıklıkla aynı tırda bulunduğu ilden başlanır.

Veriyi incelerken bazı iller için çok da fazla başka illerle tır oluşturmasının gerekmediğini gördük. Yani bazı illere giden araç sayısı oldukça fazladır. (İstanbul, Ankara..). Bu iller birkaç gün bekletilip o tırın sadece aynı ilin araçları ile oluşması beklenebilir. Ya da tam tersi bir durumda ilin zaten siparişi azdır ve yol üzerindeki bir ille birlikte gönderilmesi uygundur. Bu illeri birbirinden ayırmak için yine veri incelenerek hangi illerin başka il araçlarıyla gönderilmesi hangi araçların ise bekletilebileceği bilgisi veritabanından çıkarılır (saydırarak) ve başka bir tabloya kaydedilir ve aşağıdaki kural oluşturulur.

**Kural 2:** Tek olarak yollanan tırlara tekli (t) başka bayi ile birlikte yollanan tırlara ikili(i) dersek; eğer  $t \gg i$  ise bu il tercihen tekli gönderiliyordur, bekletilebilir. Eğer  $t/i$  yaklaşık 1 ise SRK kurallara bakılır eğer  $i \gg t$  ise tercihen başka il ile birlikte yollanıyordur, bekletmemek daha doğru bir karardır.

Aşağıdaki çizelge bazı iller için ikili,üçlü ve dördü gönderim oranlarını göstermektedir. Mesela Ankara ve İstanbul için tekli gönderimin daha yoğun olduğunu Bilecik ve Sivas için ikili ve üçlü gönderimin de yaygın olduğunu görüyoruz

Çizelge 5.1. İllerin Başka İller ile Gönderimi Oranı

İl	Tekli	İkili	Üçlü	Dördü
ANKARA	2857	183	45	6
AYDIN	360	263	16	
BİLECİK	180	195	30	2
DİYARBAKIR	229	98	27	2
EDİRNE	342	98	8	
ELAZIĞ	70	92	26	2
K.MARAŞ	39	66	11	1
KOCAELİ	829	423	19	1
MANİSA	202	294	23	1
MARDİN	118	84	27	1
MERSİN	568	180	16	
ORDU	67	39	12	3
RİZE	42	78	21	3
SİVAS	83	100	22	
İSTANBUL	6879	387	14	
İZMİR	1900	442	28	1
Ş.URFA	98	124	32	2

Aşağıda Apriori sonucunda sistemde oluşan sık kullanılan İkili Rotaları Şekil 5.2, Üçlü Rotaları Şekil 5.3 ile izleyebiliriz. Veriler incelendiğinde 5 ayrı ile uğrayacak şekilde tır oluşturulmadığı gözlenmiştir. 4 ile uğrayan tır sayısı da oldukça azdır

## İkili Rotalar

Yeni İkili Rota

Bir sayfada gösterilecek kayıt sayısını seçiniz: 50

ID	İL1	İL2	SAYI
	KOCAELİ	İSTANBUL	259
	İSTANBUL	KOCAELİ	259
	MANİSA	İZMİR	256
	İZMİR	MANİSA	256
	AYDIN	İZMİR	145
	İZMİR	AYDIN	145
	ISPARTA	ANTALYA	138
	ANTALYA	ISPARTA	138
	MUĞLA	AYDIN	128
	AYDIN	MUĞLA	128
	KOCAELİ	SAKARYA	126
	SAKARYA	KOCAELİ	126
	ADANA	MERSİN	118
	MERSİN	ADANA	118
	ESKİŞEHİR	BİLECİK	102
	BİLECİK	ESKİŞEHİR	102
	BALIKESİR	ÇANAKKALE	97
	ÇANAKKALE	BALIKESİR	97
	RİZE	TRABZON	89
	TRABZON	RİZE	89
	DÜZCE	ZONGULDAK	87
	ZONGULDAK	DÜZCE	87
	ERZURUM	SİVAS	75
	SİVAS	ERZURUM	75
	DENİZLİ	UŞAK	73
	UŞAK	DENİZLİ	73
	TEKİRDAĞ	İSTANBUL	71
	İSTANBUL	TEKİRDAĞ	71
	HATAY	ADANA	70

Şekil 5.3. İki ile Uğrayan Rotalar ve Sıklığı

## Üçlü Rotalar

Yeni Üçlü Rota

ID	IL1	IL2	IL3	SAYI
	TRABZON	RİZE	SAMSUN	15
	TRABZON	SAMSUN	RİZE	15
	RİZE	SAMSUN	TRABZON	15
	SAMSUN	TRABZON	RİZE	15
	SAMSUN	RİZE	TRABZON	15
	RİZE	TRABZON	SAMSUN	15
	GAZİANTEP	MARDİN	Ş.URFA	14
	Ş.URFA	MARDİN	GAZİANTEP	14
	Ş.URFA	GAZİANTEP	MARDİN	14
	GAZİANTEP	Ş.URFA	MARDİN	14
	MARDİN	GAZİANTEP	Ş.URFA	14
	MARDİN	Ş.URFA	GAZİANTEP	14
	BALIKESİR	MANİSA	İZMİR	12
	MANİSA	İZMİR	BALIKESİR	12
	BALIKESİR	İZMİR	MANİSA	12
	İZMİR	MANİSA	BALIKESİR	12
	İZMİR	BALIKESİR	MANİSA	12
	MANİSA	BALIKESİR	İZMİR	12
	DÜZCE	SAKARYA	ZONGULDAK	11
	DÜZCE	ZONGULDAK	SAKARYA	11
	ZONGULDAK	SAKARYA	DÜZCE	11
	SAKARYA	ZONGULDAK	DÜZCE	11
	SAKARYA	DÜZCE	ZONGULDAK	11
	ZONGULDAK	DÜZCE	SAKARYA	11
	TRABZON	ORDU	RİZE	10
	ORDU	TRABZON	RİZE	10
	RİZE	TRABZON	ORDU	10

Şekil 5.4. Üç ile Uğrayan Rotalar ve Sıklığı

Şekil 5.3 ve Şekil 5.4'deki tekrarlayan rakamlar aynı ilişki her il için tekrarlandığındandır.

SRK bulma adımı birliktelikleri bulmak için tüm veritabanını taradığından yoğun süren bir işlemdir. Ancak daha önce de bahsettiğimiz gibi bu adım sürekli çalıştırılacak bir adım olmadığı için (haftada bir, ayda bir) anlık cevap almamız gerekmez. Aşağıdaki değerler sık kullanılan rotaların bulunması adımlarının ne kadar sürdüğünü göstermektedir. Rota bulma süreleri gruplanarak hesaplanmıştır. Çünkü ikili üçlü ve dörtlü rotalar için ayrı  $min_{des}$  değeri seçilmiştir.

Çizelge 5.2. Rota Bulma Süresi

Kayıt Sayısı	İkili Rotaları Bulma Süresi (dk)	Üçlü Rotaları Bulma Süresi(dk)	Dörtlü Rotaları Bulma Süresi(dk)	Beşli Rotaları Bulma Süresi(dk)
3529	20	9	0,15	0

Yukarıdaki çizelgeyi incelediğimizde rota bulma sürelerinin kabul edilebilir zamanda tamamlandığını görüyoruz.

## 5.2. Rotalama

İkinci aşamada park alanındaki araçlar birinci aşamada bulunmuş rotalara ve kurallara göre değerini maksimize edecek ve tır maksimum dolduracak şekilde gruplara ayrılır. Modeli oluştururken aşağıdaki kurallar baz alınmıştır.

1-Tır aynı bayi araçlarından ya da aynı ile gidecek araçlardan oluşuyorsa değeri maksimumdur. (Kural1)

2- Aracın çoğunlukla tek mi yoksa başka il araçlarıyla mı gönderiliyor olduğuna göre o araçlar için tır oluşturmama kararı alınabilir(Kural2)

3- Araç toplamının tır kapasitesi kadar olması istenen durumdur. Tır kapasitesinden fazla ise yük oluşturulabilir ama artan araç olacağından oluşacak tırın değeri azdır.

4-Araç toplamı tır kapasitesinden az ise bir il daha eklenerek oluşabilecek rotalara ve o rotanın oranına bakılır. Eğer rota oranı düşükse bekletilir. Eğer hala araç sayısı az ise bir il daha eklenir.

Yukarıdaki kurallarda baz alınarak rotalamayı gerçekleştiren Şekil 4.5'deki algoritma geliştirilmiştir. Algoritmada kullanılan değişkenler Çizelge 4.2'de açıklanmıştır.

Çizelge 5.3. Model Değişkenleri ve Anlamları

Değişkenler	Anlamları
X	Tır Kapasitesi
$c_{1,2}$	Çarpan $c_1 > c_2$
K	Rota Sayısı
$FUR_{jv}$	j. Bayi ilinin bulunduğu rotalar
N	Bayi Sayısı
$B_i$	i. Bayi araç sayısı
B	Aynı tırda olma oranı
ratio(v)	v ili gezme oranı
D	Değer
$as_{jv}$	j. Bayi araç sayısı

```

1  loop i ∈ {1, 2, ..., n}
2  if (Bi ≥ x)
3    generateTransporter
4  loop j ∈ {1, 2, ..., n}
5    loop i ∈ {1, 2, ..., n}
6      if (Bi + Bj ≥ x)
7        generateTransporter
8  loop j ∈ {1, 2, ..., n}
9    loop v ∈ {1, 2, ..., k} (FURjv)
10   if ratio(v-1) >> ratio(v)
11     wait!
12   if (asjv = x)
13     d = c1*b*1/v
14     exit;
15   if (asjv > x)
16     d = c2*b*1/v
17     exit;
18   if (asjv < x)
19     loop again
20  compareValuesGenerateTransporte

```

Şekil 5.5. Rotalama Algoritması

Algoritmanın 1 ve 4. Adımlarında ilk aşamada bulduğumuz Kural1 uygulanır. Yani eğer aynı bayiye giden araçlar ya da aynı ile giden araçlar tırı dolduruyorsa bu yükler otomatik olarak oluşturulur. 8. Adımdan itibaren bulduğumuz sık kullanılan

rotalara bakarız. İlk etapta 1. Aşamada bulduğumuz iki ile uğrayıp sıklığı en yüksek olan rotalara bakarız. Sonra o illerin ikili gitme oranına bakarız eğer genelde tek yollanıyorsa bekletiriz (10. adım). Eğer ikili yollanma oranı yüksekse o ildeki bayilerin araç talebi olup olmadığına ve oluyorsa tırı tam olarak doldurup doldurmadığına bakarız. Tır kapasitesine  $x$  dersek eğer iki bayinin araç toplamı  $x$  ise tırın değeri yüksektir,  $c_1$  ile çarpılır eğer tır kapasitesini geçiyorsa daha küçük bir değer olan  $c_2$  ile çarpılır. Çünkü araç sayısının tır kapasitesinde fazla olması istenen bir durum değildir. Tır değerini hesaplarken  $b$  (Aynı Tırda Olma Oranı) ile çarpılır çünkü böylece sıklığı en yüksek olan rotanın değeri de yüksek olacaktır. Değeri hesaplarken  $1/v$  ile çarpıyor olmamızın nedeni ise tır ne kadar çok ile uğruyorsa değerinin o kadar az olacağındandır. Eğer iki uğrayarak da tır doldurulamıyorsa 3 ardından da 4 ile uğrayarak tır tamamlanmaya çalışılır.

Bir sonraki adımda bu değerleri maksimum olan rotalar ile kullanıcının önüne önerilen rotalar ve tır yapılamayan araçlar (Şekil 5.6.) listelenir.

Oluşan rota sonuçları bu işi yapan kullanıcı tarafından incelenmiş büyük oranda doğru olduğu gözlenmiştir. Ancak İstanbul araçlarında aynı yakada olma ve İzmit'e yakın olan İstanbul bayilerinin İzmit bayileriyle aynı tırda gönderilebilmesi

## Oluşturulan Rotalar

Yeni Rota

Bir sayfada gösterilecek kayıt sayısını seçiniz: 10

1	MODEL	SASI_NO	YUK_NO	İL
	225	159525	1005	AYDIN
	225	159690	1005	AYDIN
	171	6495056	1005	AYDIN
	225	159072	1005	AYDIN
	225	159781	1005	AYDIN
	110	3110494	1005	İZMİR
	172	6494359	1005	İZMİR
	225	159946	1005	AYDIN
	225	157538	1004	ADANA
	263	9026153	1004	ADANA



1 - 10 Next

## Tırı Oluşturulamayan Araçlar

MODEL	SASI_NO	İL
225	159993	ADİYAMAN
263	9025457	ADİYAMAN
172	6494565	ADİYAMAN
225	152264	AFYON
225	156995	AFYON
172	6494028	ANKARA
172	6494880	ANTALYA
199	664941	BALIKESİR
250	1791180	BALIKESİR
199	664785	BURSA

1 - 10 Next

Şekil 5.6. Rotalama Sonucu

## 6. SONUÇ

Bu tez çalışmasında, lojistikte yer alan araç rotalama problemi ayrıntılı olarak ele alınıp, bu problemin çözümü için yalnızca veri madenciliği teknikleri kullanılarak ARM (Araç Rotalama Modülü) dediğimiz bir model ortaya konmuş ve geliştirilmiştir.

Araç Rotalama Modeli iki aşamalı olarak tasarlanmıştır. İlk aşama şu ana kadar veritabanına kaydedilmiş olan sevkiyat verisinden Sık kullanılan Rota Kümesinin (SRK bulma) bulunması yani hangi bayilerin birbiri ile gittiği ilişkisinin çıkarılmasıdır. Bu ilişkiyi çıkarmak için Birliktelik Analizi algoritmalarından biri olan Apriori kullanılmıştır. İkinci aşama ise Rota Oluşturma (RO) 'dır. Rota oluşturma henüz yollanmamış araçların hangi rotalarla yollanırsa maksimum fayda olacağının tespit edildiği aşamadır. Maksimum fayda ise bizim ilk aşamada bulduğumuz rotalar ve onların kullanım sıklığı ile ilişkilidir.

Sistem tasarlanırken iki önemli kıstas göz önünde tutulmuştur. Birincisi rotalama kombinasyonel bir problem olduğundan ve şu ana kadar bulunan çözümlerdeki en büyük problem sistemlerin performansı olduğundan, performans olarak kabul edilebilir zamanda sonuç üretmesi planlanmıştır. İlk aşama SRK bulma her Rotalama işlemi yapılırken tekrarlanması gerekmediğinden bulunan SRK' lar bir veritabanı tablosuna kaydedilmiştir. Çünkü sık kullanılan rota kümesi anlık değişebilen bir küme değildir. Hatta bu küme haftalık ya da aylık periyotlarda sistemde otomatik olarak çalıştırılabilir. İkinci adım RO ise park alanındaki araç durumuna göre günde 1-2 kez çalışması gerektiğinden anlık cevap verebilecek şekilde planlanmış ve geliştirilmiştir. İkinci kıstas ise sistemin doğruluğudur. ARP bir optimizasyon problemi olduğundan ve elimizdeki problemde en doğru rota tanımı değiştiğinden kullanıcı ile birlikte sistemin doğruluğu değerlendirilmiştir. Kullanıcının buradaki değerlendirmesi standart olarak rotaların doğru bulunduğu ama detaylarda düzenlenmesi gerektiğidir. Bu düzenlemelerden birisi İstanbul ili için yaka bilgisinin de ele alınmasıdır. Ancak İstanbul Anadolu yakası ve İzmit'in ortak yük yapımında iyileştireler yapılması gerektiği gözlenmiştir. Bir de sistemdeki tasarımda üçlü ve fazlası yük yapımlarda değer az olduğundan genelde yükü bekletme ya da hiç yapmama kararı çıkmaktadır. O yüzden bu yükler genelde öneri yük olarak verilmektedir. Bir sonraki aşamada üç ve fazlası ile uğranan yüke birkaç

kural daha eklenebilir. Bu tez kapsamında bu kurallar eklenmemiştir. Çünkü eklenecek kurallar modelin genel olma özelliğini bozup daha kuruma özgü bir program haline getirmektedir.

Böyle bir sistem dağıtım ağı çok sık değişmeyen ve şu ana kadarki rotalama bilgisini tutan herhangi bir firmaya uyarlanabilir. Çözümde tüm olasılıklar yerine şu ana kadar kullanılmış olan çözümleri kullanarak seçim kümesini daraltıyor olmak ise bu sistemin avantajıdır.

## KAYNAKLAR DİZİNİ

- [1] Jiawei Han, Micheline Kamber, "Data Mining Concept and Techniues", *Morgan Kaumann Publishers*, 2005
- [2] Ayre, L., B., "Data Mining for Information Professionals", 2006.
- [3] Alex Berson, Stephen Smith, and Kurt Thearling, " An Overview of Data Mining Techniques", *Building Data Mining Applications for CRM*
- [4] Sertaç Öğüt, "Veri Madenciliği Kavramı ve Gelişim Süreci"
- [5] Kurt Thirling, "An Introduction to Data Mining Discovering Hidden Value in Your Datawarehouse"
- [6] Ali Serhan Koyuncugil, Nermin Özgülbaş, "Veri Madenciliği: Tıp ve Sağlık Hizmetlerinde Kullanımı ve Uygulamaları"
- [7] Fayyad, U., Shapiro, G., P., Smyth, P., "From Data Mining to Knowledge Discovery in Databases", *American Association for Artificial Intelligence*, 1996
- [8] A. Berson, S. Smith, K. Thearling, "Buildind Data Mining Applications for CRM", *Mcgraw Hill*, 510, USA, 1999.
- [9] Feridun Cemal Özçakır, A. Yılmaz Çamurcu, "Birliktelik Kuralı Yöntemi için bir Veri Madenciliği Yazılımı Tasarımı ve Uygulaması"
- [10] D., Bramel J., *Logic of Logistics, Theory, Algorithms & Applications for Logistics Management*, *Springer-Verlag*, New York, 1997
- [11] B. Kovalerchuk, E. Vityaev, "Data Mining in Finance: Advances in Relational And Hybrid Methods". *Kluwer Academic Publishers*, 308, USA. 2001.
- [12] Fisher M. L., "Optimal Solution of Vehicle Routing Problems Using Minimum K-trees", *Operations Research* 42, 1994, 626-642.
- [13] Toth, P. and D. Vigo, "Models, relaxations and exact approaches for the capacitated vehicle routing problem". *Discrete Applied Mathematics*, vol.123, no.1-3, pp.487-512, 2002.
- [14] P. M. Thompson and H. N. Psaraftis, "Cyclic Transfer Algorithms for the Multivehicle Routing and Scheduling Problems", *Operations Research* 41:935-946. 1993.
- [15] G. A. P. Kinderwater and M. W. P. Savelsbergh, "Vehicle Routing: Handling Edge Exchanges". In E. H. L. Aarts and J. K. Lenstra (eds), *Local Search in Combinatorial Optimization* Wiley, Chichester. 1997.

- [16] M. L. Fisher and R. Jaikumar, "A Generalized Assignment Heuristic for Vehicle Routing", *Networks*, 11:109-124, 1981.
- [17] É. D. Taillard. "Parallel Iterative Search Methods for Vehicle Routing Problems". *Networks* 23, 661-673. 1993.
- [18] J. Kelly and J. P. Xu, "A Network Flow-Based Tabu Search Heuristic for the Vehicle Routing Problem". *Transportation Science*, 30:379-393. 1996.
- [19] D. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning". *Addison Wesley Publishing Company Inc*, New York. 1989.
- [20] K. A. De Jong, "An Analysis of the Behavior of a Class of Genetic Adaptive Systems". Ph.D. thesis, *University of Michigan*, U.S.A. 1975.
- [21] J. H. Holland, "Adaptation in Natural and Artificial Systems", *University of Michigan Press*, Ann Arbor. 1975.
- [22] Xiangpei Hu, Minfang Huang, "An Intelligent Solution System for a Vehicle Routing Problem in Urban Distribution", *International Journal of Innovative Computing, Information and Control*, Volume 3, Number 1, February 2007. pp.189-198
- [23] Hrvoje Marković, Ivana Čavar, Tonči Carić, "Using Data Mining To Forecast Uncertain Demands in Stochastic Vehicle Routing Problem", *ISEP 2005*
- [24] H. G. Santos a, L. S. Ochi a, E. H. Marinho a, L. M.A. Drummond, "Combining an Evolutionary Algorithm with Data Mining to Solve a Single-Vehicle Routing Problem", Preprint submitted to *Elsevier Science*, 24 January 2006
- [25] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules", In Proc. 20th Int. Conf. Very Large Data Bases, VLDB, pages 487–499. *Morgan Kaufmann*, 1994.
- [26] R. Agrawal and R. Srikant, "Mining sequential patterns", In *Proc. 11th Int. Conf. on Data Engineering*, pages 3–14, 1995.
- [27] Han, J., Kamber, M., "Data Mining: Concepts and Techniques", *Morgan Kaufmann Publishers Inc.*, 2001.
- [28] Berry, M., J., A., Linoff, G., S., "Data Mining Techniques for Marketing, Sales, and Customer Relationship Management", Second Edition, *Wiley Publishing*, 2004.
- [29] Agrawal, R. ve Srikant, R., "Fast Algorithms for Mining Association Rules", In Proceedings of the 20th *International Conference on Very Large Databases (VLDB '94)*, Santiago, Chile, 487-489, 1994.

- [30] Frawley, W. J., Piatetsky-Shapiro, G. ve Matheus, C. J., “Knowledge Discovery Databases: An Overview, in Knowledge Discovery in Databases”, *AAAI 58 AI Magazine*, Cambridge, 1-27, 1991.
- [31] Agrawal R. ve Srikant R., “Mining Sequential Patterns”, 11th *International Conference on Data Engineering*, Taipei, Taiwan, 3-14, 1995.
- [32] Zhu, H., “On-Line Analytical Mining of Association Rules”, MSc. Thesis, *Simon Fraser University*, Ottawa, Canada, 1998.
- [33] Houtsma, M. ve Swami, A, “Set-Oriented Mining for Association Rules in Relational Databases”, Proceedings of the 11th *IEEE International Conference on Data Engineering*, Taipei, Taiwan, 25-34, 1995.
- [34] Das A., Ng, W. K. ve Woon Y. K. , “Rapid Association Rule Mining”, In Proceedings of the Tenth International Conference on Information and Knowledge Management, ACM Press, Atlanta, GA, USA, 487-499, 2001.
- [35] Zaki, M. J. ve Hsiao, C. J., “CHARM: An Efficient Algorithm for Closed Itemset Mining”, In 2nd *SIAM International Conference on Data Mining (SDM'02)*, Eds. Grossman, R. L., Han, J., Kumar, V., Mannila, H. ve Motwani, R., Siam, Arlington, VA, USA, 457-473, 2002.
- [36] Gao, W., “A Hierarchical Document Clustering Algoritm”, MSc Thesis, *Dalhousie University*, Halifax, Nova Scotia, 2004.
- [37] Sever, H. ve Oğuz, B., “Veritabanlarında Bilgi Keşfine Formal Bir Yaklaşım, Kısım 1: Eşleştirme Sorguları ve Algoritmalar”, *Bilgi Dünyası*, 3, 2, Ekim, 173-204, 2002.
- [38] Simchi-Levi D., Bramel J., Logic of Logistics, Theory, Algorithms & Applications for Logistics Management, *Springer-Verlag*, New York, 1997
- [39] <http://neo.lcc.uma.es/radiaeb/WebVRP/>, z.t.:20.11.2008
- [40] Rushton A., Croucher P., Baker P., Handbook of Logistics and Distribution Management 3rd Edition, Kogan Page, Limited, 2006

## ÖZGEÇMİŞ

1985 yılında Bilecik’ de doğdu. Lise eğitimini Bilecik Refik Arslan Öztürk Fen Lisesi’nde tamamlayarak 2002 yılında Hacettepe Üniversitesi Bilgisayar Mühendisliği Bölümü’ne girdi. 2006 yılında Bilgisayar Mühendisi unvanı ile mezun oldu. 2007 yılı güz döneminde Gebze Yüksek Teknoloji Enstitüsü Bilgisayar Mühendisliği Bölümü’nde yüksek lisans eğitimine başladı. 2007’den beri özel bir firmada bilgisayar mühendisi olarak çalışmaktadır.