İSTANBUL TEKNİK ÜNİVERSİTESİ ★FEN EDEBİYAT FAKÜLTESİ
MATEMATİK MÜHENDİSLİĞİ PROGRAMI

NAIVE BAYES YÖNTEMİ İLE BANKACILIK SEKTÖRÜ ÜZERİNE BİR VERİ MADENCİLİĞİ UYGULAMASI

BİTİRME ÖDEVİ

Ece Hazal AYDIN 090080013
Büşra ÇELİKKARAOĞLU 090080078

Teslim Tarihi: 17.05.2013

Tez Danışmanı: Yar. Doç. Dr. Ahmet KIRIŞ

MAYIS 2013

ÖNSÖZ

Bu çalışmayı hazırlamamızda bize yol gösteren, sabrını, yardımını ve bilgisini hiçbir zaman esirgemeyen Sayın Hocamız Yar. Doç. Dr. Ahmet KIRIŞ'a, öğrenim hayatımız boyunca her türlü desteğini aldığımız değerli arkadaşlarımıza, bize sevgiyi, güveni veren ve bugünlere gelmemizi sağlayan ailelerimize en içten teşekkürlerimizi sunarız.

Mayıs, 2013

Ece Hazal AYDIN

Büşra ÇELİKKARAOĞLU

İÇİNDEKİLER

ÖZET

Bu çalışmada özel bir Portekiz bankasının, müşterilerine banka kredisi satması üzerine bir veri madenciliği uygulaması yapılmıştır. Bu uygulama ile bankanın geçmiş kampanyalarına katılım bilgileri bilinen müşterilerin, aynı bankanın gelecek kampanyasına katılım durumlarının belirlenmesi amaçlanmaktadır.

Bu amaç ile banka için hazırlanmış olan, müşteri kişisel bilgilerini, kampanya kapsamında müşterilerin aranma sürelerini, kampanyaya katılım durumlarını içeren veriler "http://archive.ics.uci.edu/ml/index.html" sitesinden alınmıştır. Müşterilerin gelecek kampanyaya katılım durumlarının belirlenmesi için ise sınıflandırma modelinin algoritmalarından Naive Bayes algoritması seçilmiştir.

Veri madenciliği uygulamasını gerçekleştirmek için Oracle veritabanının 11g 2. Sürümü (11g R2) ve bunun üzerine Oracle Data Mining ile ilgili paket programlar yüklenmiştir. Daha sonra, müşteri bilgilerini içeren tablo veritabanına aktarılmış ve gerekli modeller oluşturulmuştur. Bu modeller yardımıyla en doğru tahminler elde edilmeye çalışılmış ve sonuçlar yorumlanmıştır.

1. GİRİŞ

Her geçen gün gelişen teknoloji ile bilgisayar sistemleri insan yaşamının ayrılmaz bir parçası haline gelmeye başlamıştır. Günümüzde her alanda kullanılan farklı donanım ve yazılıma sahip bilgisayarlar, her an yeni oluşan verileri kaydetmektedirler. Veritabanı teknolojilerinde sağlanan gelişmelerle de verilerin saklanması daha kolay bir hal alırken, silinmesi ya da kaybolması ihtimali imkansıza yaklaşmıştır. Bu şekilde durmadan büyüyen ve işlenmediği sürece değersiz gibi görünen veri yığınları oluşmaktadır. Çok fazla miktarda verinin toplanması, kişilerin ve kurumların bilgi sistemleri içindeki istenilen asıl bilgilere ulaşmasını zorlaştırmaktadır. Veri madenciliği ile büyük veritabanları içerisindeki verilerin matematiksel modelleme kullanan bilgisayar programları ile analiz edilerek anlamlı bilgilere dönüştürülmesi sağlanmaktadır. Veri madenciliğinin en çok kullanıldığı sektörler olarak sağlık, biyoloji ve genetik, endüstri ve mühendislik, pazarlama, bankacılık, sigortacılık ve borsa, kriminoloji ve iletişim ön plana çıkmaktadır.

Bu bitirme projesi kapsamında; veri madenciliği uygulaması ile özel bir Portekiz bankasının müşterilerinin geçmiş yıllardaki banka kredisi kampanyalarına katılım durumları üzerine model oluşturulmuş ve herhangi bir müşterinin aranması durumunda gelecek kampanyaya katılım durumunun ne olacağı tahmin edilmeye çalışılmıştır. Bir müşterinin gelecek kampanyaya katılımını belirlerken hangi aşamalardan geçildiği ve veri madenciliği uygulamasına neden gereksinim duyulduğu bu tez kapsamında anlatılmıştır.

2. VERİ MADENCİLİĞİ

Günümüzde kullanılan veritabanı yönetim sistemleri eldeki verilerden sınırlı çıkarımlar yaparken, geleneksel çevrimiçi işlem sistemleri (on-line transaction processing systems) de bilgiye hızlı ve güvenli erişimi sağlamaktadır. Fakat ikisi de eldeki verilerden analizler yapıp anlamlı bilgiler elde etme imkanını sağlamakta yetersiz kalmışlardır. Verilerin yığınla artması ve anlamlı çıkarımlar elde etme ihtiyacı arttıkça, uzmanlar Knowledge Discovery in Databases (KDD) adı altındaki çalışmalarına hız kazandırmışlardır. Bu çalışmalar sonucunda da veri madenciliği (Data Mining) kavramı doğmuştur. Veri madenciliğinin temel amacı, çok büyük veri tabanlarındaki ya da veri ambarlarındaki veriler arasında bulunan ilişkiler, örüntüler, değişiklikler, sapma ve eğilimler, belirli yapılar gibi bilgilerin matematiksel teoriler ve bilgisayar algoritmaları kombinasyonları ile ortaya çıkartılması ve bunların yorumlanarak değerli bilgilerin elde edilmesidir [1].

2.1 Tanım

İlişkisel veritabanı sistemleriyle ulaşılan veriler tek başına bir anlam ifade etmezken veri madenciliği teknolojisi bu verilerden anlamlı bilgi üretilmesinde öncü rol oynamaktadır. Veri madenciliği; matematiksel yöntemler yardımıyla, biriken veri yığınları içerisinde bulunan verilerin birbirleriyle ilişkisini ortaya çıkartmak için yapılan analiz ve kurulan modeller sonucunda elde edilecek bilgi keşfi sürecidir.

2.2 Kullanım Alanları

Günümüzde veriler çok hızlı bir şekilde toplanabilmekte, depolanabilmekte, işlenebilmekte ve bilgi olarak kurumların hizmetine sunulabilmektedir. Bilgiye hızlı erişim, firmaların sürekli yeni stratejiler geliştirip etkili kararlar almalarını sağlayabilmektedir. Bu süreçte araştırmacılar, büyük hacimli ve dağınık veri setleri üzerinde firmalara gerekli bilgi keşfini daha hızlı gerçekleştirebilmeleri için veri madenciliği üzerine çalışmalar yapmaktadırlar. Tüm bu çalışmalar doğrultusunda veri madenciliği günümüzde yaygın bir kullanım alanı bulmuştur. Veri madenciliği,

perakende ve pazarlama, bankacılık, sağlık hizmetleri, sigortacılık, tıp, ulaştırma, eğitim, ekonomi, güvenlik, elektronik ticaret alanında yaygın olarak kullanılmaktadır.

2.3 Temel Veri Madenciliği Problemleri ve Çözüm Yöntemleri

Veri madenciliği uygulaması gerektiren problemlerde, farklı veri madenciliği algoritmaları ile çözüme ulaşılmaktadır. Veri madenciliği görevleri iki başlık altında toplanmaktadır.

- Eldeki verinin genel özelliklerinin belirlenmesidir.
- Kestirimci/Tahmin edici veri madenciliği görevleri, ulaşılabilir veri üzerinde tahminler aracılığıyla çıkarımlar elde etmek olarak tanımlanmıştır.

Veri madenciliği algoritmaları aşağıda açıklanmaktadır.

2.3.1 Karakterize etme (Characterization)

Veri karakterizasyonu hedef sınıfındaki verilerin seçilmesi, bu verilerin genel özelliklerine göre karakteristik kuralların oluşturulması olayıdır.

2.3.2 Ayrımlaştırma (Discrimination)

Belirlenen hedef sınıfa karşıt olan sınıf elemanlarının özellikleri arasında karşılaştırma yapılmasını sağlayan algoritmadır. Karakterize etme metodundan farkı mukayese yöntemini kullanmasıdır.

2.3.3 Sınıflandırma (Classification)

Sınıflandırma, veri tabanlarındaki gizli örüntüleri ortaya çıkarabilmek için veri madenciliği uygulamalarında sıkça kullanılan bir yöntemdir. Verilerin sınıflandırılması için belirli bir süreç izlenir. Öncelikle var olan veritabanının bir kısmı eğitim amaçlı kullanılarak sınıflandırma kurallarının oluşturulması sağlanır. Bu kurallar kullanılarak veriler sınıflandırılır. Bu veriler sınıflandırıldıktan sonra eklenecek veriler bu sınıflardan karakteristik olarak uygun olan sınıfa atanır. Sınıflandırma problemleri için "Oracle Data Mining" (ODM)' ın uyumlu olduğu çözüm yöntemleri Naive Bayes (NB), Karar Destek Vektörleri (SVM), Karar Ağaçları'dır [1].

2.3.4 Tahmin etme (Prediction)

Kayıt altında tutulan geçmiş verilerin analizi sonucu elde edilen bilgiler gelecekte karşılaşılacak aynı tarz bir durum için tahmin niteliği taşıyacaktır. Bu tür problemler için Oracle Data Mining (ODM) de kullanılan regresyon analizi yöntemi SVM'dir.

2.3.5 Birliktelik kuralları (Association rules)

Birliktelik kuralları gerek birbirini izleyen gerekse de eş zamanlı durumlarda araştırma yaparak, bu durumlar arasındaki ilişkilerin tanımlanmasında kullanılır. Bu modelin yaygın olarak Market Sepet Analizi uygulamalarında kullanıldığı bilinmektedir.

2.3.6 Kümeleme (Clustering)

Yapı olarak sınıflandırmaya benzeyen kümeleme metodunda, birbirine benzeyen veri grupları aynı tarafta toplanarak kümelenmesi sağlanır. Sınıflandırma metodunda sınıfların kuralları, sınırları ve çerçevesi bellidir ve veriler bu kriterlere göre sınıflara atanırken, kümeleme metodunda sınıflar arası bir yapı ortaya çıkartılıp, benzer özellikte olan verilerle yeni gruplar oluşturmak asıl hedeftir. Verilerin kendi aralarındaki benzerliklerinin göz önüne alınarak gruplandırılması, yöntemin pek çok alanda uygulanabilmesini sağlamıştır.

2.3.7 Aykırı değer analizi (Outlier analysis)

İstisnalar veya sürpriz olarak tespit edilen aykırı veriler, bir sınıf veya kümelemeye tabii tutulamayan veri tipleridir. Aykırı değerler bazı uygulamalarda atılması gereken değerler olarak düşünülürken bazı durumlarda ise çok önemli bilgiler olarak değerlendirilebilmektedirler.

2.3.8 Zaman serileri (Time series)

Yapılan veri madenciliği uygulamalarında kullanılan veriler çoğunlukla statik değildir ve zamana bağlı olarak değişmektedir. Bu metot ile bir veya daha fazla niteliğin belirli bir zaman aralığında, eğilimindeki değişim ve sapma durumlarını inceler. Belirlenen zaman aralığında ölçülebilir ve tahmin edilen/beklenen değerleri karşılaştırmalı olarak inceler ve sapmaları tespit eder.

2.3.9 Veri görüntüleme (Visualization)

Bu metot, çok boyutlu özelliğe sahip verilerin içerisindeki karmaşık bağlantıların/bağıntıların görsel olarak yorumlanabilme imkanını sağlar. Verilerin birbirleriyle olan ilişkilerini grafik araçları görsel ya da grafiksel olarak sunar. ODM histogramlar ile bu metodu desteklemektedir.

2.3.10 Yapay sinir ağları (Artificial neural networks)

Yapay sinir ağları insan beyninden esinlenilerek geliştirilmiş, ağırlıklı bağlantılar aracılığıyla birbirine bağlanan işlem elemanlarından oluşan paralel ve dağıtılmış bilgi işleme yapılarıdır. Yapay sinir ağları öğrenme yoluyla yeni bilgiler türetebilme ve keşfedebilme gibi yetenekleri hiçbir yardım almadan otomatik olarak gerçekleştirebilmek için geliştirilmişlerdir. Yapay sinir ağlarının temel işlevleri arasında veri birleştirme, karakterize etme, sınıflandırma, kümeleme ve tahmin etme gibi veri madenciliğinde de kullanılan metotlar mevcuttur.

2.3.11 Genetik algoritmalar (Genetic algorithms)

Genetik algoritmalar doğada gözlemlenen evrimsel sürece benzeyen, genetik kombinasyon, mutasyon ve doğal seçim ilkelerine dayanan bir arama ve optimizasyon yöntemidir. Genetik algoritmalar parametre ve sistem tanılama, kontrol sistemleri, robot uygulamaları, görüntü ve ses tanıma, mühendislik tasarımları, yapay zeka uygulamaları, fonksiyonel ve kombinasyonel eniyileme problemleri, ağ tasarım problemleri, yol bulma problemleri, sosyal ve ekonomik planlama problemleri için diğer eniyileme yöntemlerine kıyasla daha başarılı sonuçlar vermektedir.

2.3.12 Karar ağaçları (Decision trees)

Ağaç yapıları esas itibariyle kural çıkarma algoritmaları olup, veri kümelerinin sınıflanması için "if-then" tipinde kullanıcının rahatlıkla anlayabileceği kurallar inşa edilmesinde kullanılırlar. Karar ağaçlarında veri kümesini sınıflamak için "Classification and Regression Trees (CART)" ve "Chi Square Automatic Interaction Detection (CHAID)" şeklinde iki yöntem kullanılmaktadır.

2.3.13 Kural çıkarma (Rules induction)

İstatistiksel öneme sahip "if-else" kurallarının ortaya çıkarılması problemlerini inceler.

3. TEMEL KAVRAMLAR VE MATEMATİKSEL ALTYAPI

Bu tez kapsamında, özel bir Portekiz bankasının müşterilerinin, geçmiş kampanyalara katılım durumlarından yararlanarak sıradaki kampanyaya katılım durumlarını tahmin etme probleminin çözümü amacıyla Naive Bayes algoritması kullanılacağından bu yöntem anlatılacaktır.

3.1 Naive Bayes Yöntemi

Naive Bayes sınıflandırma problemlerinin çözümünde kullanılan bir algoritmadır. Temel olarak olasılık teorisini kullanmaktadır. Bu bölümde önce yöntemin teorisi, sonrasında da yöntemle ilgili iki örnek verilmiştir.

3.1.1 Temel kavramlar

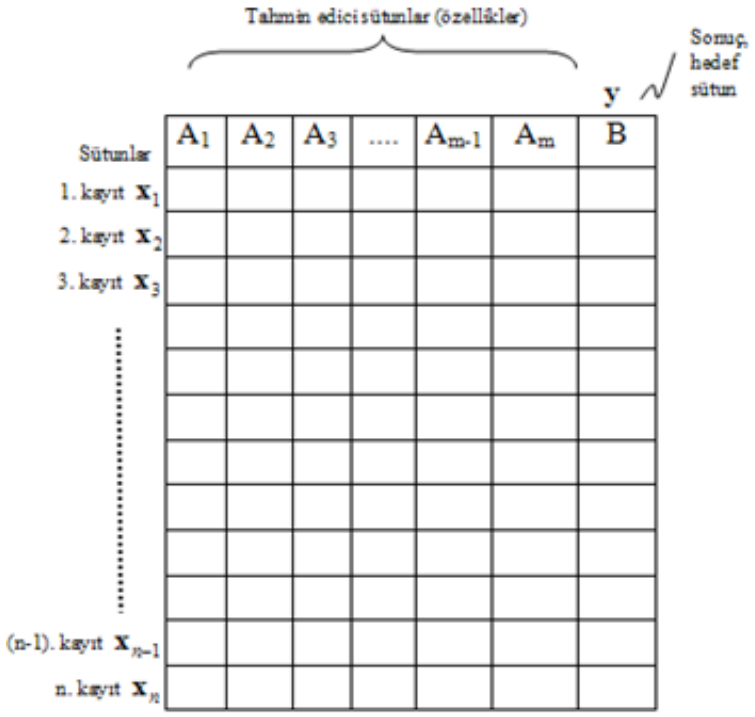
Olasılık: Bir olayın gerçekleşme oranıdır, [0-1] arasında değer alabilir, $P(A)$ ile gösterilir ve

$P(A)=1$, A olayının mutlaka gerçekleşeceğini

$P(A)=0$, A olayının gerçekleşmesinin mümkün olmadığını ifade eder.

Vektör: Burada kullanacağımız anlamıyla bir satır vektör $\mathbf{x}=\{x_1, x_2, x_3, ..., x_{m-1}, x_m\}$ şeklinde *m* elemanı ile belirlenen ve i. elemanı x_i ile verilen bir büyüklüktür.

Veri madenciliği uygulanacak olan veri kümesi Şekil 3.1 formatındadır. Tabloda her satır (her kayıt) bir vektör ($\mathbf{x}_i$) olarak düşünülür, $\mathbf{x}_i$ vektörünün j. elemanı i. kaydın A_j sütunundaki değerine karşı gelir. Son sütun (B) yani $\mathbf{y}$ vektörü, veri madenciliği ile tahmin edilmek istenen hedef özelliktir. Dolayısıyla *n* kayıt ve ($m+1$) sütundan oluşan bir tabloda her biri *m* boyutlu *n* tane belirleyici $\mathbf{x}_i$ vektörü ve bir tane hedef sütun (B), yani $\mathbf{y}$ vektörü vardır.



Şekil 3.1: Bir veri kaydı örneği

3.1.2 Teori

Naive Bayes yöntemi ile sınıflandırma koşullu olasılık hesabına dayanmaktadır. Şekil 3.1 de görüldüğü üzere tüm değerleri belirli geçmiş bir veri kümesinde, B yani sonuç sütunu, diğer $A_i, (i = 1,...,m)$ sütunlarına bağlı kabul edilerek, $P(B = b_j \mid A_i = a_{ik},...(i = 1,...,m))$, olasılıkları hesaplanır, burada $j = 1,...,s$ ve $k = 1,...,m_i$ dir. Bu ifade ile her biri m_i tane farklı gruptan oluşan A_i sütunları a_{ik} değerlerini aldıklarında, bu A_i sütunlarına bağlı olarak, B sütununda bulunan s tane farklı grubun b_j değerlerinden her birini alma olasılıkları hesaplanmaktadır. Geçmiş veri kümesi yardımıyla hesaplanan bu olasılıklar, yeni gelecek verinin hangi gruba dahil edileceğinin, yani B sütununun tahmininde kullanılacaktır.

Konuyu anlaşılır kılmak için, tahmin edici sütun önce bir tane, A_1, sonra iki tane, A_1, A_2 alınarak, B sütununun bunlara bağlı olasılıkları hesaplanarak problem basitleştirilmiş daha sonra ise m sütun alınarak problem genelleştirilmiştir.

Bu amaçla öncelikle koşullu olasılık kavramının açıklanması gerekmektedir. A ve B iki olay olmak üzere, bu olayların olma olasılıkları $P(A)$ ve $P(B)$ ile gösterilir.

Eğer A ve B olaylarının gerçekleşmesi birbirine bağlı değilse, bu iki olayın birlikte olma olasılığı

$$P(A,B) = P(A) \times P(B) \tag{3.1}$$

ile verilir. Örneğin A olayı, o gün havanın yağmurlu olması ve B olayı ise atılan bir madeni paranın yazı gelme olasılığı ise, bu iki olay birbirinden bağımsızdır ve bu iki olayın birlikte olma olasılıkları her bir olayın olma olasılıklarının çarpımına eşittir.

Eğer A ve B olayları birbirine bağlı ise, bu iki olayın birlikte olma olasılıkları; A' nın olma olasılığı ile A biliniyorken B' nin olma olasılığının çarpımı ile yani

$$P(A,B) = P(A)\,P(B \mid A) \tag{3.2}$$

veya B' nin olma olasılığı ile B biliniyorken A' nın olma olasılığının çarpımı ile yani

$$P(A,B) = P(B)\,P(A \mid B) \tag{3.3}$$

ile verilir. Dolayısıyla buradan (3.2) ve (3.3) denklemleri birbirine eşitlenerek, A biliniyorken B' nin olma olasılığı

$$P(B \mid A) = \frac{P(B)P(A \mid B)}{P(A)} \tag{3.4}$$

ile verilir. Örneğin A olayı havanın yağmurlu olması, B olayı ise Ali' nin balığa çıkma olayı ise, B olayının A olayına bağlı olduğu açıktır ve A biliniyorken B' nin olma olasılığı yani hava yağmurlu iken Ali' nin balığa çıkma olayı (3.4) ifadesiyle hesaplanır.

Bir olayın olması ve olmaması olasılıkları toplamı $P(B) + P(B^{\perp}) = 1$ dir. Burada "$^{\perp}$" üst indisi B olayının değilini göstermektedir. Dolayısıyla Ali hava yağmurlu iken balığa çıktığı gibi, yağmur yağmazken de balığa çıkabilir, yani bir B olayına bağlı olarak A olayının olma olasılığı

$$P(A) = P(A,B) + P(A,B^{\perp}) = P(B)P(A \mid B) + P(B^{\perp})P(A \mid B^{\perp}) \tag{3.5}$$

şeklinde verilir. Bu ifade, (3.4)' te kullanılırsa,

$$P(B \mid A) = \frac{P(B)P(A \mid B)}{P(B)P(A \mid B) + P(B^{\perp})P(A \mid B^{\perp})} \tag{3.6}$$

elde edilir. Eğer A ve B olayları farklı değerler alabiliyorsa, örneğin Ali' nin balığa çıkması (b_1), işe gitmesi (b_2), spor yapması (b_3) gibi üç farklı B olayı varsa bu

durumda $P(B=b_1)+P(B=b_2)+P(B=b_3)=1$ dir. (3.5) ifadesine benzer bir şekilde bu kez A olayı r tane ayrık a_k ve B olayı s tane ayrık b_j değeri alıyorsa;

$$P\left(A=a_k\right)=\sum_{j=1}^{s}P\left((A=a_k),(B=b_j)\right)=\sum_{j=1}^{s}P\left(B=b_j\right)P\left((A=a_k)\,|\,(B=b_j)\right) \tag{3.7}$$

elde edilir. (3.7) ifadesi (3.4)' te yerine yazıldığında ise,

$$P\left((B=b_j)\,|\,(A=a_k)\right)=\frac{P\left(B=b_j\right)P\left((A=a_k)\,|\,(B=b_j)\right)}{\sum_{k=1}^{s}P\left(B=b_k\right)P\left(A\,|\,(B=b_k)\right)} \tag{3.8}$$

elde edilir. (3.8) ifadesinin A ve B olaylarının ikiden fazla değer alabildikleri durum için (3.6) ifadesinin genelleştirilmiş hali olduğu açıktır. Bu ifade Şekil.3.1' de verilen tabloda B sonuç sütununu tahmin edici tek bir A_1 sütunu olması halinde B sütununun alabileceği değerlerin olasılıklarının hesaplanmasında kullanılır. Ancak gerçek hayatta sadece biri tahmin edici, diğeri hedef sütun olmak üzere iki sütun olması değil, hedef sütunu tahmin edici birçok sütun bulunması beklenir.

Bu nedenle (3.8) ifadesinde A gibi sadece bir tahmin edici sütun yerine m tane A_i sütunu olduğunu ve bunların her birinin r_i tane bağımsız değer alabildiği yani örneğin A_1 sütunu $r_1=5$, A_2 sütunu $r_2=3$ farklı değer alabildiğini varsayalım. Bu durumda (3.8) ifadesinde A yerine $A_1, A_2, ..., A_m$ gibi m tane olay alınırsa;

$$P\left(B=b_j\,|\,A_1=a_{1j_1},A_2=a_{2j_2},...,\,A_m=a_{mj_m}\right)= \frac{P\left(B=b_j\right)P\left(A_1=a_{1j_1},A_2=a_{2j_2},...,\,A_m=a_{mj_m}\,|\,B=b_j\right)}{\sum_{k=1}^{s}P\left(B=b_k\right)P\left(A_1=a_{1j_1},A_2=a_{2j_2},...,\,A_m=a_{mj_m}\,|\,B=b_k\right)} \tag{3.9}$$

ifadesi elde edilir. Tahmin edici her sütunun yani her olayının birbirinden bağımsız olduğu kabulü yapılırsa, sonuç olarak

$$P(B=b_k\,|\,A_1=a_{1j_1},A_2=a_{2j_2},...,\,A_m=a_{mj_m})=\frac{P(B=b_k)\times\prod_{i=1}^{m}P(A_i=a_{ij_i}\,|\,B=b_k)}{\sum_{\forall r|b_r\in B}\left(P(B=b_r)\times\prod_{i=1}^{m}P(A_i=a_{ij_i}\,|\,B=b_r)\right)} \tag{3.10}$$

ifadesi elde edilir. Burada $j_i = 1,...,m_i$ ve $k = 1,...,s$ için bu olasılık değerleri hesaplanmalıdır, ayrıca $\forall r \,|\, b_r \in B$ terimi hedef sütunun alabileceği tüm farklı değerler üzerinde toplam alınacağını ifade etmektedir [2].

3.1.3 Örnekler

Örnek 1) *Tek boyut için:* Yapılan bir anket sonucunda 200 kişinin eğitim durumları "ortaokul", "lise" ve "üniversite" olarak gruplanmış ve eğitim durumlarına göre çocuk sayıları ikinci bir sütunda Tablo 3.1a' daki gibi belirtilmiştir. Oracle Data Mining (ODM) verilerin olasılık hesaplarını arka planda otomatik olarak işleyip kullanıcıya sadece sonucu bildirir. Yapılan anketin sonuçları kullanılarak ODM'nin arka planda olasılıkları nasıl hesapladığı açıklanmıştır. Kısaltma amacıyla Eğitim Durumu=ED, Ortaokul=O, Lise=L, Üniversite=U, Çocuk Sayısı=CS, Bir=B, İki=I ve Yok=Y şeklinde sembolize edilecektir. Tablo 3.1a verisinden elde edilen her farklı gruptaki kişi sayısı Tablo 3.1b ile gösterilmiştir.

Tablo 3.1a: Eğitim Durumu - Çocuk Sayısı ilişkisi

Eğitim Durumu	**Çocuk Sayısı**
Ortaokul	İki
Ortaokul	Yok
Lise	Bir
Üniversite	Yok
Üniversite	İki
. .	. .

Tablo 3.1b: Her gruptaki kişi sayısı

ED	**CS**=İki	**CS**=Bir	**CS**=Yok
Ortaokul	5	15	20
Lise	50	25	15
Üniversite	40	18	12

Tablo 3.1b yardımıyla sözü edilen olasılıklar (3.8) ifadesi kullanılarak;

$$P(CS = B|ED = O) = \frac{P(CS = B)P(ED = O\backslash CS = B)}{(P(CS = B)P(ED = O\backslash CS = B) + P(CS = I)P(ED = O\backslash CS = I) + P(CS = Y)P(ED = O\backslash CS = Y)} \quad (3.11)$$

$$= \frac{\frac{58}{200}\times\frac{15}{58}}{\frac{58}{200}\times\frac{15}{58}+\frac{95}{200}\times\frac{5}{95}+\frac{47}{200}\times\frac{20}{47}} = \frac{15}{40} \quad (3.12)$$

$$P(CS = I \mid ED = O) = 0.125 \quad (3.13)$$

$$P(CS = B \mid ED = O) = 0.375 \quad (3.14)$$

$$P(CS = Y \mid ED = O) = 0.5 \quad (3.15)$$

olarak hesaplanmıştır. Beklendiği üzere bu olasılıkların toplamı "1" etmektedir. Diğer durumlar için olasılıklar yine Tablo 3.1b den kolay bir şekilde hesaplanabilir. Diğer olasılıklar;

$$P(CS = I \mid ED = L) = 0.5556 \qquad P(CS = I \mid ED = U) = 0.5714$$

$$P(CS = B \mid ED = L) = 0.2778 \qquad P(CS = B \mid ED = U) = 0.2571$$

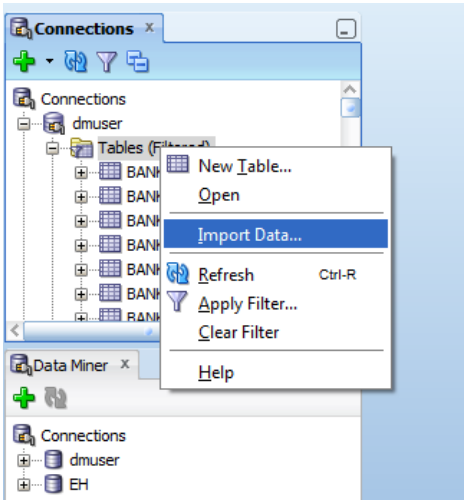
$$P(CS = Y \mid ED = L) = 0.1666 \qquad P(CS = Y \mid ED = U) = 0.1714 \quad (3.16)$$

bulunmuştur. Bu olasılıklar, eğitim durumu bilinen birinin çocuk sayısı hakkında tahmin yapabilmeyi sağlayacaktır. Örneğin, eğitim durumu ortaokul olan birinin çocuk sayısının bir olması beklenirken, eğitim durumu üniversite olan birinin çocuk sayısının bu olasılıklara göre iki olması beklenecektir. Tek boyutlu veri tabloları için bir kestirimci özellik olduğundan el ile olasılık hesabı yapmak zor değildir. Fakat hem hedef özelliğin ikiden fazla hem de kestirimci özellik sayısının birden fazla olduğu durumlarda tablodan okuma zorlaşacaktır ve yukarıdaki formülün uygulanması gerekecektir. Oracle Data Mining (ODM), Naive Bayes yöntemini kullanarak model oluşturmayı sağlar ve bu olasılıkları otomatik olarak hesaplayıp ve tahmin yapılmasını kolaylaştırır.

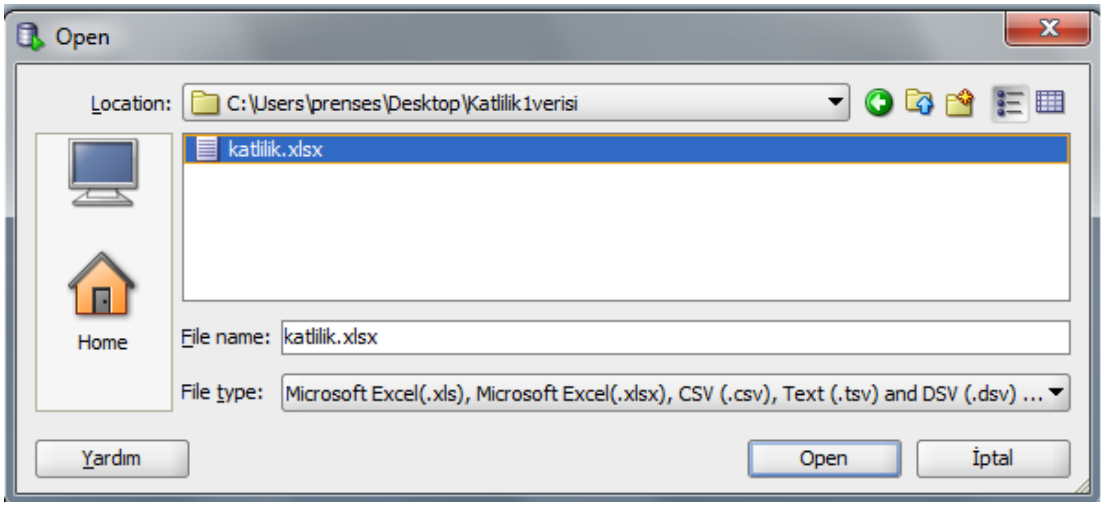
Aynı işlem Oracle Data Mining 11gR2 kullanılarak yapılmıştır:

1. ODM uygulaması için kullanılacak tablo, Oracle SQL Developer arayüzünde de yer alan "Connections" altındaki dmuser kullanıcısı üzerinde, "Tables" sağ

tıklanıp Şekil 3.2 deki gibi "Import Data" seçilir ve verinin Excel dosyası Şekil 3.3 de gösterildiği şekilde seçilerek tablo oluşturulmaya başlanır.

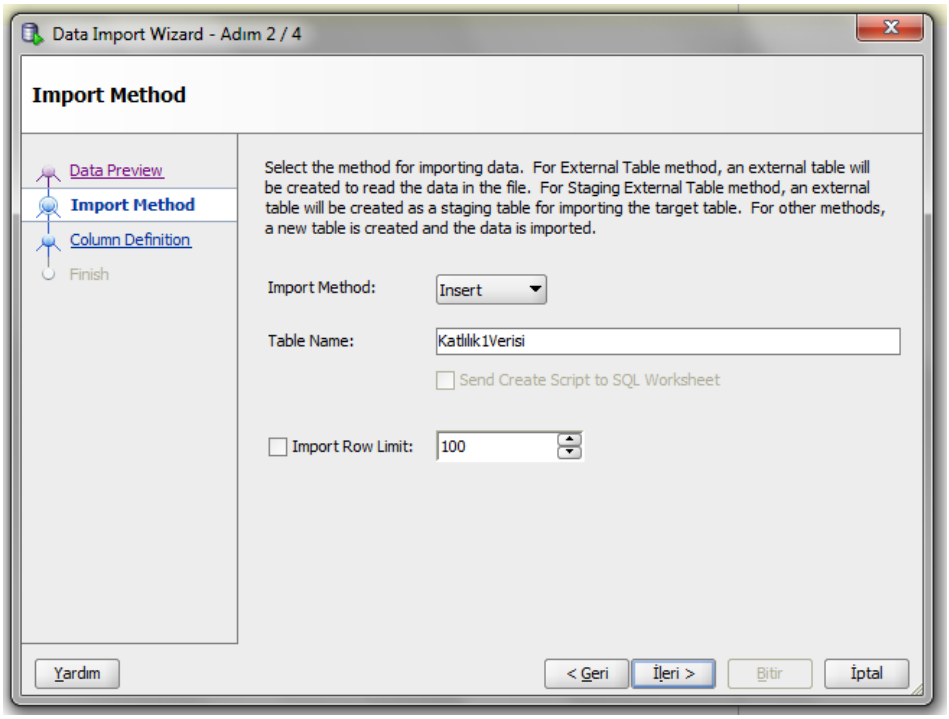


Şekil 3.2: Tablo oluşturma (1.Aşama)



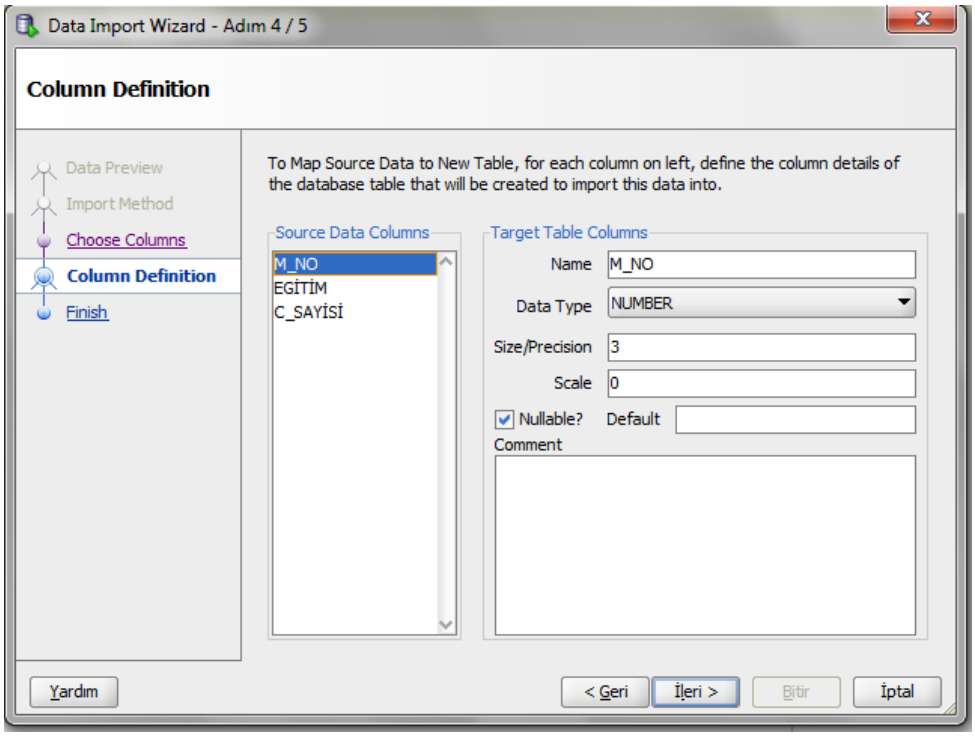
Şekil 3.3: Tablo oluşturma (2.Aşama)

2. "Data Import Wizard" kısmında ikinci adımda "Table Name" olarak "Katlilik1Verisi" verilmiştir (Şekil 3.4).



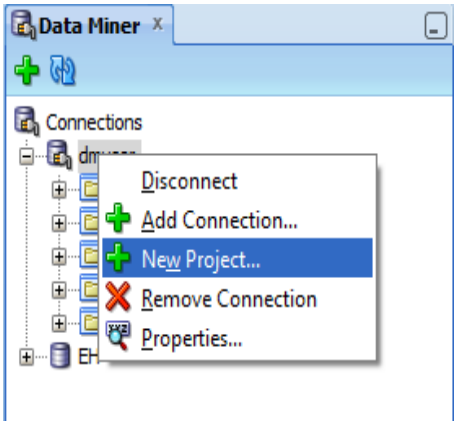
Şekil 3.4: Tabloyu isimlendirme

3. "Data import wizard" kısmında üçüncü adımda tabloya eklenecek verilerin her biri için veri tipleri ve büyüklükleri verinin aldığı değerlere göre belirlenmiş ve tablo oluşturulmuştur (Şekil 3.5).

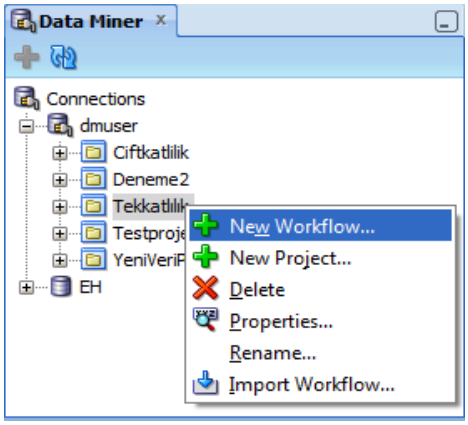


Şekil 3.5: Tablo sütunlarının düzenlenmesi

4. Data Mining kullanıcısı olan dmuser içinde şekil 3.6 de ki gibi “New Project” ile ‘Tekkatlılık’ adında proje oluşturulmuş ve şekil 3.7 de görüldüğü gibi “New Workflow” sekmesiyle uygulamanın yer alacağı bir iş akış diyagramı eklenmiştir.

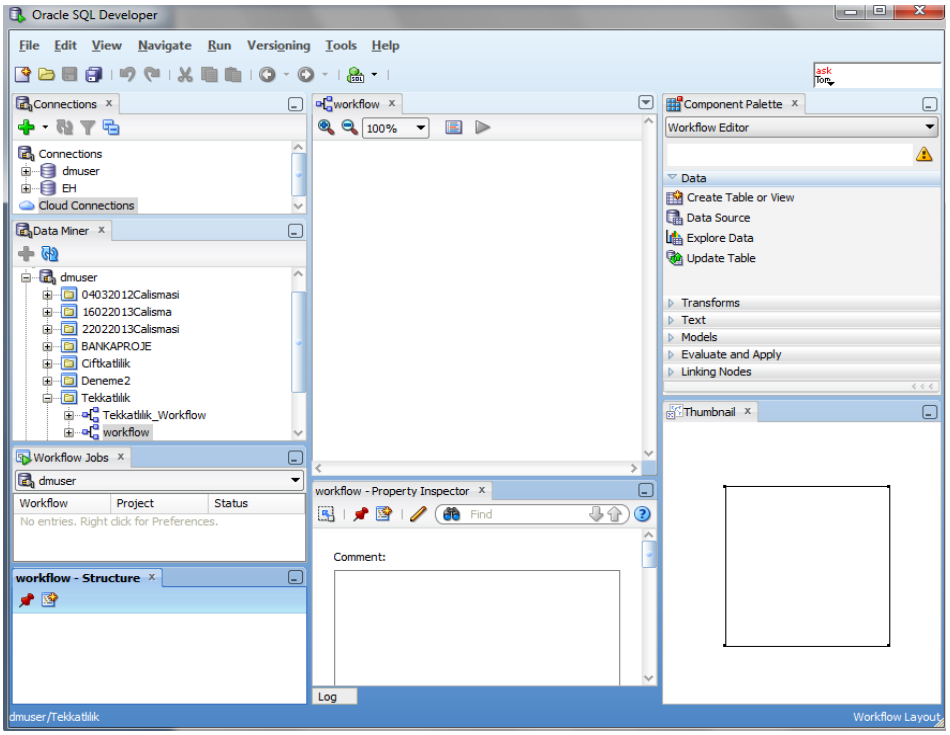


Şekil 3.6: Proje Oluşturma



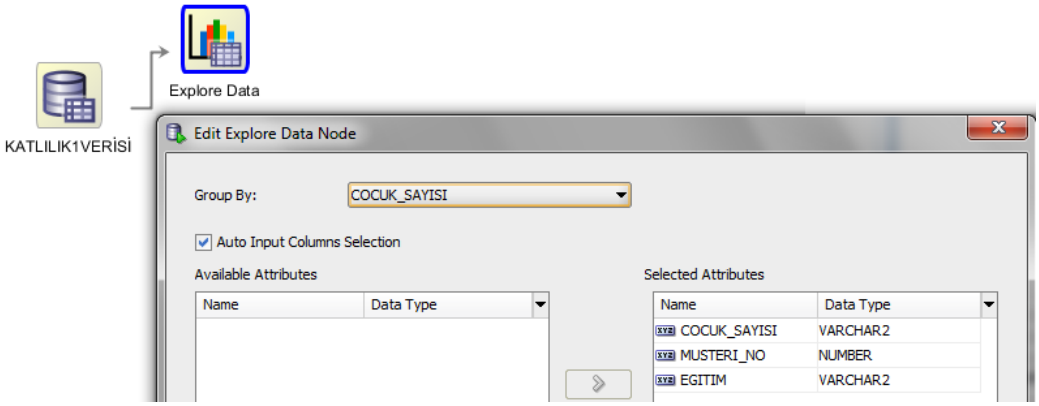
Şekil 3.7: Workflow oluşturma

5. Oluşturulan iş akış diyagramı şekil 3.8 de görülmektedir. Oracle SQL Developer arayüzünde “Component Palette” yer alır. Bu kısımda uygulama için kullanılabilecek tüm düğümler bulunmaktadır. İlk olarak “Data Source” düğümü diyagram içine yerleştirilmiş ve olasılığı el ile hesaplanan, 200 kişinin eğitim ve çocuk sayılarını içeren ‘Katlılık1Verisi’ adlı veri gelen ekrandan seçilmiştir.



Şekil 3.8: ODM genel görünüm

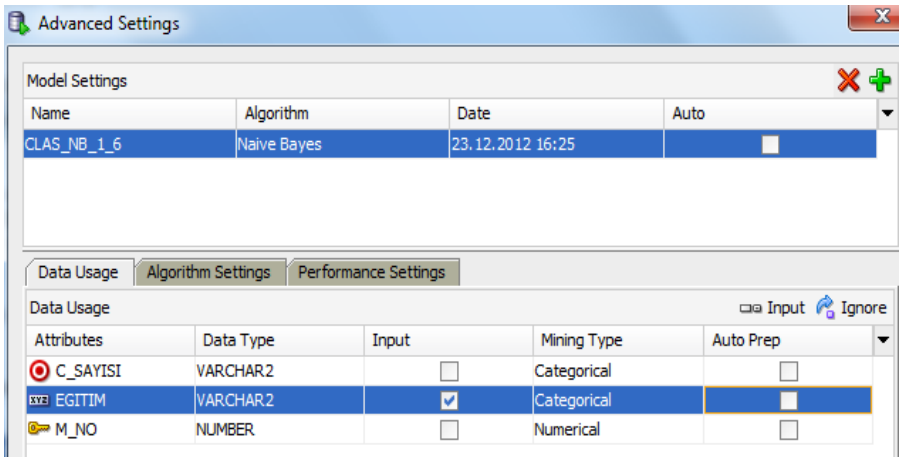
6. Veri seçildikten sonra, Component Palette bölümündeki “Explore Data” düğümü diyagramda yer alan veri ile bağlanmış, özelliklerinden “Group By” kısmında target (hedef) seçilmiştir. Burada kişilerin çocuk sayılarının tahmin edilmesi amaçlandığından, verinin hedef sütununa göre gruplama yapılması için Şekil 3.9 de ‘COCUK_SAYISI’ seçilmiştir ve düğüm çalışılmıştır.



Şekil 3.9: Explore data ayarları

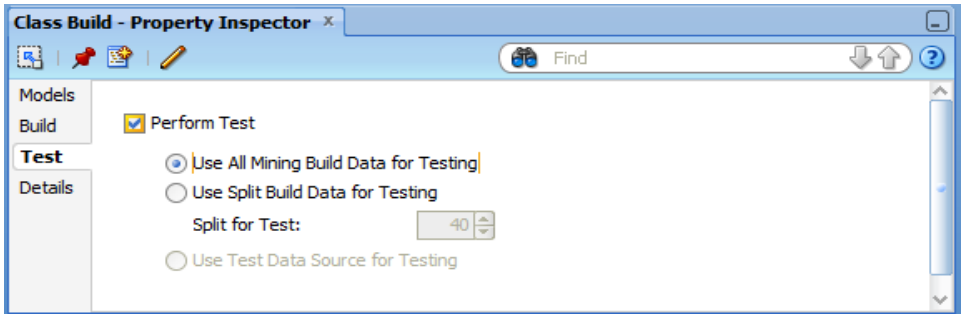
İşlem tamamlandığında düğüm üzerinde “View Data” seçilerek “Statics” bölümünde veri ile ilgili istatistikler incelenebilir.

7. Component Palette kısmında “Models” sekmesi altında yer alan “Classification” düğümü diyagramdaki veri ile bağlanmış ve özellikleri ayarlanmıştır. Sınıflandırma işleminde sadece Naive Bayes algoritması kullanılmıştır. “Target” yani hedef sütun “COCUK_SAYISI”, “CaseID” ise birincil anahtar olan “MUSTERI_NO” seçilmiştir. Naive-Bayes algoritmasında ayarların otomatik olarak yapılmaması için “Auto” seçeneğini kaldırılmıştır. Ayarların düzenlenmesi için “Advanced Settings” bölümünde “Data Usage” sekmesinde “Input” ve “Ignore” ile girdi değerleri düzenlenmiştir. Burada “CaseID” ve tahmin edilecek sütun olan “Target” ın girdi olmamasına dikkat edilmiştir (Şekil 3.10). Girdi olarak alınan “EGITIM” sütunu için “Auto Prep (Auto Preparation)” seçeneği kaldırılmıştır. Böylece veri üzerinde Naive Bayes algoritmasının herhangi bir hazırlık yapmasına izin verilmemiştir. Ayrıca veri ağırlıklarının tabloda olduğu gibi yer alması için “Performance Settings” sekmesinde “Natural” seçilerek ayarlar tamamlanmıştır.



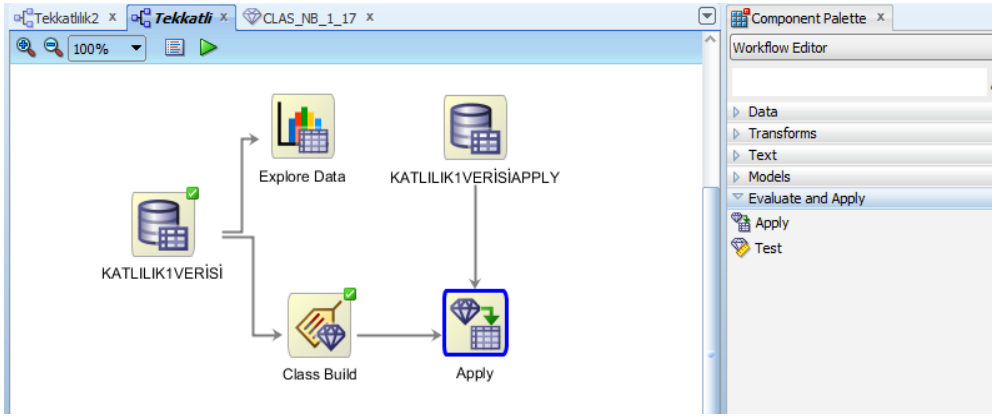
Şekil 3.10: Model Oluşturma (1.Aşama)

8. "Property Inspector" panosunda "Test" bölümünde "Use All Mining Build Data for Testing" seçilip (Şekil 3.11) "Class Build" düğümü çalıştırılmıştır.



Şekil 3.11: Model Oluşturma (2.Aşama)

Veri üzerinde tahmin yapabilmek için "Class Build" modeli oluşturulmuştur. Bu model kullanılarak "Apply" düğümü ile birlikte tahminler ve olasılık değerleri incelenebilecektir. Burada tahmin için aynı veri ele alınmıştır. Veri ve "Class Build" adlı model "Apply" düğümü ile Şekil 3.12 de gösterildiği gibi bağlanıp çalıştırılmıştır. Tahmin sonuçları olasılıklarla birlikte Şekil 3.13 deki gibi görülmüştür. Gerçek bir uygulamada ODM'de model oluşturulurken "Apply" düğümünde kullanılan veri "Class Build" düğümünde kullanılandan farklı olmalıdır. Çünkü "Apply" aktivitesindeki amaç hedef sütunu hiç olmayan ancak aynı tahmin edici sütunlara sahip farklı veriler içeren yeni veri için tahmin sütununun oluşturulmasıdır. Fakat burada ODM kullanmadan bulunan sonuçların doğruluğunun görülmesi amaçlandığından "Apply" düğümü aynı tabloya uygulanmıştır.



Şekil 3.12: Modelin Uygulanması

ekkatlik2 × Apply ×

Columns | SQL

View: Actual Data | Sort... | Filter: Enter Where Clause

	MUSTERI_NO	EGITIM	COCUK_SAYISI	CLAS_NB_1_15_PRED	CLAS_NB_1_15_PROB
1	1	universite	iki	iki	0,5714
2	2	universite	iki	iki	0,5714
3	3	universite	iki	iki	0,5714
4	4	universite	iki	iki	0,5714
5	5	universite	iki	iki	0,5714
80	80	lise	iki	iki	0,5556
81	81	lise	iki	iki	0,5556
82	82	lise	iki	iki	0,5556
83	83	lise	iki	iki	0,5556
84	84	lise	iki	iki	0,5556
85	85	lise	iki	iki	0,5556
180	180	ortaokul	bir	yok	0,5
181	181	ortaokul	yok	yok	0,5
182	182	ortaokul	yok	yok	0,5
183	183	ortaokul	yok	yok	0,5
184	184	ortaokul	yok	yok	0,5
185	185	ortaokul	yok	yok	0,5

Şekil 3.13: Örnek 1'in ODM'de tahmin ve olasılık sonuçları

Şekil 3.13 de görüldüğü gibi "CLAS_NB_1_15_PRED" isimli sütunda eğitim durumlarına göre çocuk sayısı tahminleri yapılmıştır. Formül kullanarak bulunan sonuçlar ile "CLAS_NB_1_15_PROB" isimli sütundaki olasılıklar karşılaştırıldığında hesaplanan ve ODM ile elde edilen olasılık değerlerinin aynı olduğu görülmüştür. Sonuçların aynılığı çok daha büyük veriler için doğrudan ODM ile model oluşturulup, güvenilir tahminler yapılabileğini göstermektedir.

ODM, Naive Bayes yöntemiyle sınıflandırılması mümkün olan her durum için olasılıkları hesaplayarak bir model oluşturup, bu modeli yukarıdaki gibi aynı tablo veya yeni tabloların hedef sütunu üzerinde tahmin yapmak için kullanmaktadır. Modelin doğruluğunun test edilmesi amacıyla formüllerle yapılacak işlemlerde aynı veri ve hesaplanan olasılıklar kullanılarak yeni tahmin tablosu oluşturulmuştur. Örneğin eğitim durumu ortaokul olan bir kişinin çocuğunun olmama olasılığı 0,5 olarak hesaplandığı için tahmin "yok" ve sonucun güvenirliliği 0,5'dir. Eğitim durumu lise olan bir kişinin bir çocuğunun olma olasılığı ise 0,4444 olduğu için modelin tahmini "iki" ve sonucun güvenilirliği 0,5556'dır. Bu şekilde işleme devam edilerek tüm tablo oluşturulmuştur.

Tablo 3.2: Örnek 1 için elde edilen olasılık sonuçları

Eğitim Durumu	Çocuk Sayısı	Tahmin	Güvenilirlik
Ortaokul	İki	Yok	0.5
Ortaokul	Yok	Yok	0.5
Lise	Bir	İki	0.5556
Üniversite	Yok	İki	0.5714
Üniversite	İki	Bir	0.2571
.	.	.	.

Modelin tüm güvenilirliği ise gerçek değerler ile tahmini değerlerin karşılaştırılması sonucu elde edilmiştir ve sonuçlara Tablo 3.2 de yer verilmiştir. Tablo 3.3 deki güvenilirlik matrisin değerleri "Apply" düğümünün sonuçlarına dayanılarak yazılmıştır.

Tablo 3.3: Güvenilirlik Matrisi

	Bir	İki	Yok
Bir	0	43	15
İki	0	90	5
Yok	0	27	20

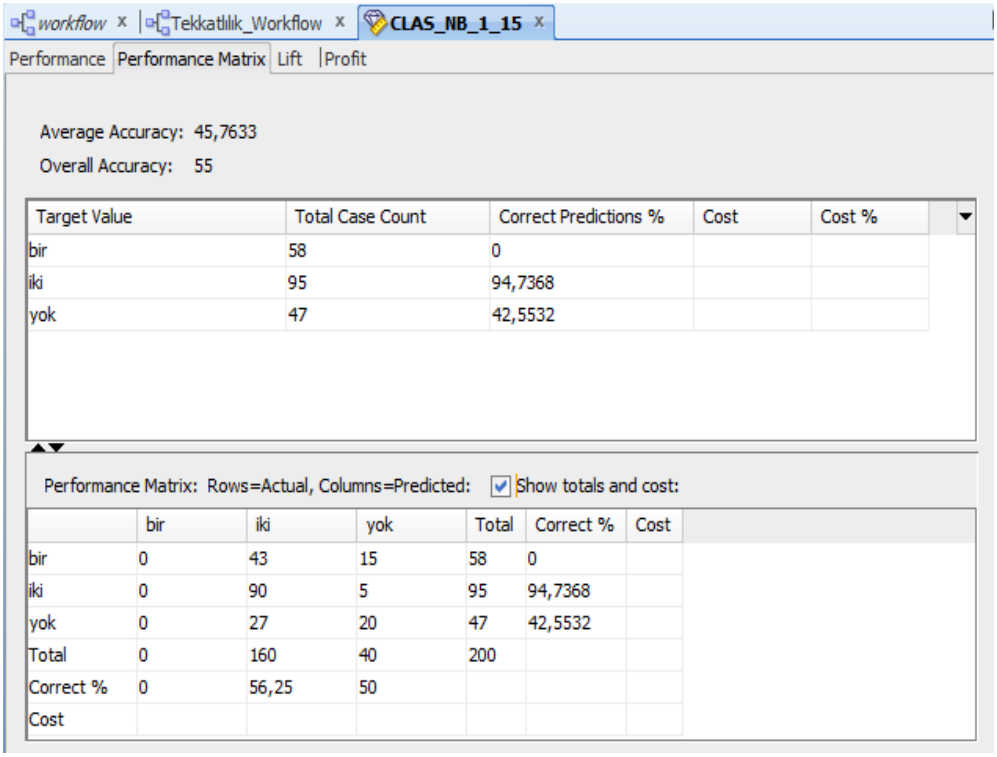
Tablo 3.3 de görülen güvenilirlik matrisinde satırlar gerçek değerleri, sütunlar ise tahmin sonuçlarını gösterir. Örneğin gerçekte çocuk sayısı iki iken, modelin de iki (doğru) olarak tahmin ettiği kayıt sayısı 90, gerçekte çocuk sayısı yok iken modelin

iki olarak tahmin ettiği kayıt sayısı (yanlış) 27 dir. Dolayısıyla matrisin köşegeni doğru kayıt sayısını, köşegen dışı ise yanlış tahmin edilen kayıt sayısını gösterir.

Buradan modelin doğruluğu;

$$MD = \frac{0+90+20}{(0+43+15)+(0+90+5)+(0+27+20)} = \frac{110}{200} = 0.55 \qquad (3.17)$$

olarak elde edilmiştir. Modelin doğruluğu ODM kullanılarak daha kolay hesaplanmıştır. "Class Build" düğümünü çalıştırdıktan sonra "View Test Result" seçilerek şekil 3.14 deki "Performance Matrix" kısmından modelin doğruluğu incelenebilir.



Şekil 3.14: Örnek 1'in "Performance matrix" sonucu

"Overall Accuracy" sonucu modelin doğruluğunu göstermektedir. Build, test ve apply işlemlerinde aynı veri kullandığından el ile hesaplanılan sonuçlarla ODM'nin sonuçları aynı çıkmıştır.

Örnek 2) *İki Boyut için:* Yapılan bir anket sonucunda 100 kişinin eğitim durumları "Ortaokul", "Lise" ve "Üniversite", arabalarının olup olmamaları ise "Evet" ve "Hayır" olarak belirlenmiş ve bu kişilerin çocuk sayıları "Bir", "İki" ve "Yok" şeklinde Tablo 3.4 deki gibi belirtilmiştir. Kayıtların dağılımı Tablo 3.5 te

verilmiştir. Burada kısaltma amacıyla Eğitim_Durumu=ED, Ortaokul=O, Lise=L, Üniversite=U, Çocuk_Sayısı=CS, Bir=B, İki=I, Yok=Y, Araba=A, Evet=E ve Hayır=H şeklinde sembolize edilmiştir.

Tablo 3.4: Model Verisi

Eğitim Durumu	**Araba**	**Çocuk Sayısı**
Üniversite	Hayır	Bir
Lise	Evet	Yok
Lise	Evet	İki
Ortaokul	Hayır	Yok
Ortaokul	Hayır	Bir
. .	. .	. .

Tablo 3.5: Kayıt Dağılımları

Eğitim Durumu	**Araba**	**Çocuk Sayısı**	**Kayıt Sayısı**
Üniversite	Hayır	İki	6
Üniversite	Evet	İki	2
Lise	Hayır	İki	11
Lise	Evet	İki	4
Ortaokul	Hayır	İki	5
Ortaokul	Evet	İki	20
Üniversite	Hayır	Bir	1
Üniversite	Evet	Bir	5
Lise	Hayır	Bir	2
Lise	Evet	Bir	0
Ortaokul	Hayır	Bir	0
Ortaokul	Evet	Bir	12
Üniversite	Hayır	Yok	3
Üniversite	Evet	Yok	0
Lise	Hayır	Yok	7
Lise	Evet	Yok	16
Ortaokul	Hayır	Yok	2
Ortaokul	Evet	Yok	4

Tablo 3.5 yardımıyla ankete katılan kişilerin eğitim durumları ile arabasının olup olmaması verisi kullanılarak çocuk sayıları (3.10) denklemiyle belirlenmeye çalışılmıştır. Burada iki belirleyici özellik olduğundan formül bu tabloya uygun formda yazılmıştır. Denklem, Tablo 3.5 için uygulandığında,

$$P(CS=B|ED=U,A=E)=\frac{P(CS=B)P(E=U\backslash CS=B)P(A=E\backslash CS=B)}{P(CS=B)P(E=U\backslash CS=B)P(A=E\backslash CS=B)+P(CS=I)P(E=U\backslash CS=I)P(A=E\backslash CS=I)+P(CS=Y)P(E=U\backslash CS=Y)P(A=E\backslash CS=Y)} \quad (3.18)$$

$$=\frac{\frac{20}{100}\times\frac{6}{20}\times\frac{17}{20}}{\frac{20}{100}\times\frac{6}{20}\times\frac{17}{20}+\frac{48}{100}\times\frac{8}{48}\times\frac{26}{48}+\frac{32}{100}\times\frac{3}{32}\times\frac{20}{32}}=\frac{0.051}{0.11305}=0.451$$

sonucu elde edilmiştir. Bu sonuca göre eğitim durumu üniversite ve arabası olan kişilerin %45.1 olasılıkla çocuk sayısı "bir" olacaktır. Benzer şekilde diğer olasılıklardan bazıları hesaplanırsa;

$$P(CS=I\,|\,ED=U,\ A=E)=0.38301 \qquad P(CS=B\,|\,ED=U,\ A=E)=0.451$$

$$P(CS=Y\,|\,ED=L,\ A=E)=0.594 \qquad P(CS=Y\,|\,ED=L,\ A=H)=0.5459$$

$$P(CS=I\,|\,ED=O,\ A=H)=0.7389 \qquad P(CS=I\,|\,ED=O,\ A=E)=0.4926$$

(3.19)

Olasılıkların uzun işlemlerle hesaplanıp, tahminde bulunulması yerine, aynı işlem ODM 11gR2 üzerinde Örnek 1 de belirtilen adımlarla yapılmıştır. Örnek 2 nin apply işleminin sonucu tahminler ve olasılıklar ile birlikte Şekil 3.15 de gösterilmiştir.

	MUSTERI_NO	EGITIM_DURUMU	ARABA	COCUK_SAYISI	CLAS_NB_1_16_PRED	CLAS_NB_1_16_PROB
1	13	universite	evet	bir	bir	0,451
2	12	universite	evet	bir	bir	0,451
3	11	universite	evet	bir	bir	0,451
4	10	universite	evet	bir	bir	0,451
5	9	universite	evet	bir	bir	0,451
6	2	universite	evet	iki	bir	0,451
14	71	ortaokul	evet	iki	iki	0,4926
15	98	ortaokul	evet	yok	iki	0,4926
16	73	ortaokul	evet	iki	iki	0,4926
54	32	lise	hayir	iki	yok	0,5459
55	33	lise	hayir	bir	yok	0,5459
56	34	lise	hayir	bir	yok	0,5459
57	51	lise	hayir	yok	yok	0,5459
85	16	universite	hayir	yok	iki	0,6442
86	15	universite	hayir	yok	iki	0,6442
87	14	universite	hayir	bir	iki	0,6442
88	8	universite	hayir	iki	iki	0,6442
98	82	ortaokul	hayir	iki	iki	0,7389
99	99	ortaokul	hayir	yok	iki	0,7389
100	100	ortaokul	hayir	yok	iki	0,7389

Şekil 3.15: Örnek 2 için tahmin ve olasılık değerleri

Ayrıca Örnek 2 deki modelin doğruluğu Şekil 3.16 te gösterilmiştir.

ciftkatlilik | CLAS_NB_1_16

Performance | Performance Matrix | Lift | Profit

Average Accuracy: 53,8194

Overall Accuracy: 59

Target Value	Total Case Count	Correct Predictions %	Cost	Cost %
bir	20	25		
iki	48	64,5833		
yok	32	71,875		

Performance Matrix: Rows=Actual, Columns=Predicted: Show totals and cost:

	bir	iki	yok	Total	Correct %	Cost
bir	5	13	2	20	25	
iki	2	31	15	48	64,5833	
yok	0	9	23	32	71,875	
Total	7	53	40	100		
Correct %	71,4286	58,4906	57,5			
Cost						

Şekil 3.16: Örnek 2 nin "Performance Matrix" sonucu

4. UYGULAMA VE SONUÇLAR

Bu bölümde Naive Bayes algoritmasının Oracle Veri Madenciliği ile uygulanışı anlatılmıştır. Tez kapsamındaki problem, Portekizli bir bankacılık kurumunun doğrudan pazarlama kampanyaları ile ilgili olarak müşteriler ile telefon görüşmelerinde en fazla sayıda olumlu yanıt alabilmesi ile ilgilidir. Bu tez kapsamında, modelin oluşturulması, uygulanması ve test süreçleri Naive Bayes yöntemiyle anlatılmıştır.

4.1. Naive Bayes Yönteminin Uygulanması

Bu bölümde "http://archive.ics.uci.edu/ml/index.html" sitesinden alınan verinin incelenmesi yapılmıştır. Veri içindeki tablolar sırasıyla müşterilerin yaş, meslek, medeni hali, eğitim durumu, ödenmeyen kredi bilgisi, kendilerine ait bakiyeleri, ev kredisi alıp almadıkları, bireysel krediye sahip olup olmadıkları, müşteriyle hangi kanaldan iletişime geçildiği, müşteriyle hangi gün, hangi ay, ne kadar süre konuşulduğu, en son ne zaman konuşulduğu, müşterinin kaç kampanyaya katıldığı, en son konuşmada kampanyaya katılımı sağlanıp sağlanmadığı bilgileri vardır. Tablo 4.1 de tablolardaki veri içerikleri hakkında bilgi verilmiştir.

Tablo 4.1: Tablo İçeriği

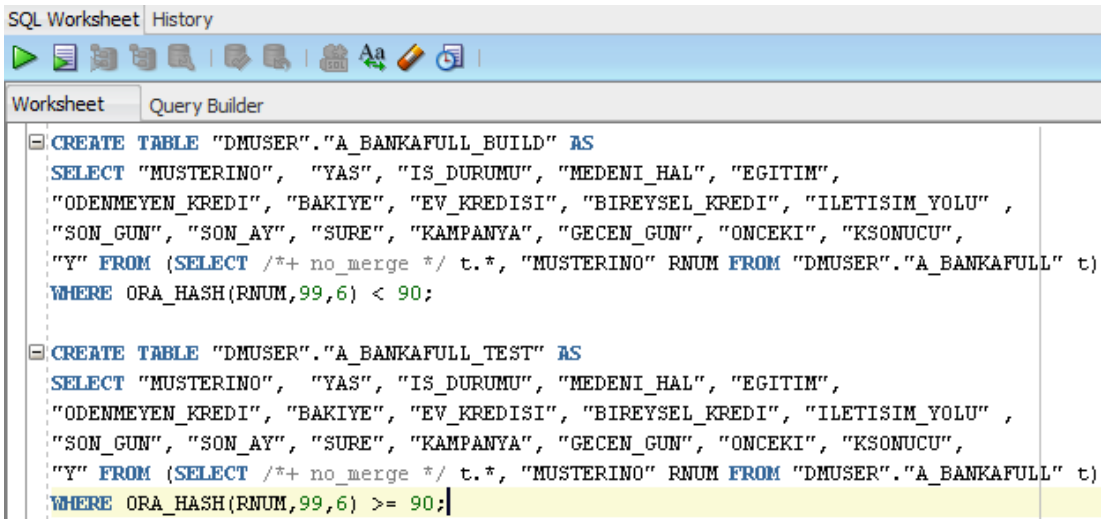
Sütun Adı	Sütun Açıklaması
musterino	Müşteri numarasını belirtir.
yas	Müşterilerin yaşını belirtir. 18 ile 95 arasında değerler alır.
is_durumu	Müşterinin meslek bilgisi bulunur.
medeni_hal	Müşterinin bekar, evli veya boşanmış olduğu bilgisini içerir.
egitim	Müşterinin hesabındaki para miktarını belirtir. Negatif değer alabilir.
odenmeyen_kredi	Müşterinin ödenmeyen kredisi bulunup, bulunmadığı bilgisini içerir. Evet, hayır değerleri alır.
bakiye	Müşterinin hesabındaki para miktarını belirtir. Negatif değer alabilir.

ev_kredisi	Müşterinin ev kredisi olup olmadığını belirtir. Evet, hayır değerleri alır.
bireysel_kredi	Müşterinin bireysel kredisi olup olmadığını belirtir. Evet, hayır değerleri alır.
iletisim_yolu	Müşteriyle hangi yoldan iletişime geçildiğini belirtir. 3 değer alır: Bilinmiyor, cep telefonu, telefon.
son_gun	Müşteriyle en son hangi gün iletişime geçildiği bilgisi içerir.
son_ay	Müşteriyle en son hangi ay iletişime geçildiği bilgisi içerir.
sure	Müşteriyle en son iletişimin ne kadar sürdüğü bilgisini içerir.
kampanya	Müşterinin en son kampanya için kaç kere arandığı bilgisini içerir.
gecen_gun	Müşteri ile en son iletişim kurulmasının üzerinden kaç gün geçtiği bilgisini içerir.
onceki	Müşteri ile en son kampanyadan önce kaç defa iletişime geçildiği bilgisini içerir.
ksonucu	Müşterinin bir önceki kampanyaya katılmasına ikna edilip edilmediği bilgisini içerir. 4 farklı değer alır: Bilinmiyor, başarılı, başarısız, diğer.
Y (Hedef)	Müşterinin kampanyaya dahil olup olmadığı bilgisini verir. Evet veya hayır şeklinde iki değer alır.

4.2. Veri Tablosunun Hazırlanması

Veri, Örnek 1'in ODM üzerinde uygulanışında anlatıldığı gibi veritabanına aktarılmıştır. Model oluşturulurken eldeki verinin tamamı model oluşturma amacıyla kullanılmamalı, modelin doğruluğunu test etmek için verinin bir kısmı ayrılmalıdır. Eğer model, eldeki verinin tümüyle oluşturulursa, modelin doğruluğunu yanıltıcı şekilde arttıracağı açıktır. Bu yüzden, toplam 45211 kayıt içeren veri üzerinde model oluşturmadan önce verinin %10 luk kısmı test için ayrılır. Test verisi ayrıldıktan sonra, elde kalan büyük veri ile model oluşturulur. Daha sonra, oluşturulan modelin doğruluğu test verisi üzerinden incelenir. Test sonucunda tatminkar sonucun alınması durumunda model, hedef kolonu bilinmeyen yeni müşteriler için kullanılabilir. Test sonucunun istenildiği kadar yüksek çıkmaması durumunda model, yeni sütunlar eklenerek veya farklı algoritmalar kullanılarak yeniden oluşturulur, istenilen sonuca ulaşılıncaya kadar aynı adımlar tekrarlanır.

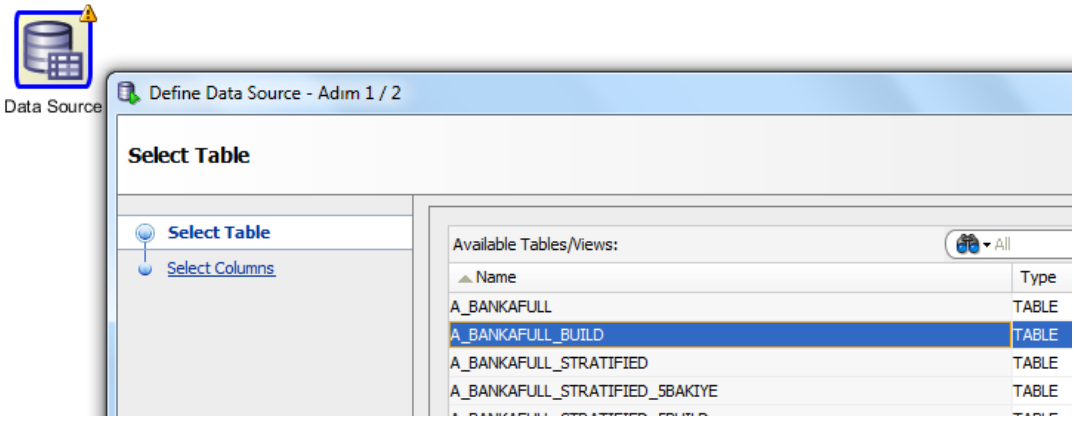
İncelenen veriyi "model oluşturma" ve "test" verisi olarak ikiye ayırmak için, bir SQL Worksheet penceresi açılıp, şekil 4.1 deki SQL komutu kullanılmıştır.



Şekil 4.1: Veriyi ayıran SQL komutu

4.3 Model Oluşturma

1. "Component Palette" kısmından "Data Source" düğümü iş diyagramına alındıktan sonra veri olarak daha önceden oluşturulmuş olan "A_BANKAFULL_BUILD" tablosu olarak seçilmiştir (Şekil 4.2).



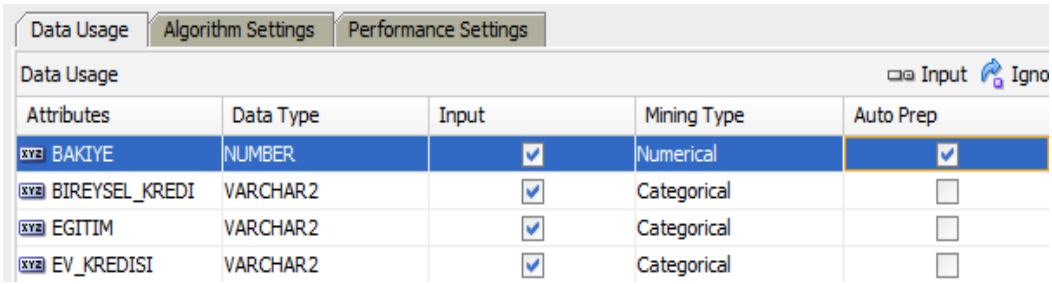
Şekil 4.2: Veri Seçimi

2. Verinin her sütununda kaç farklı veri olduğunun incelenmesi için "Explore Data" düğümü kullanılır. Bunun sonucuna göre BAKIYE sütunu için 1338 farklı, GECEN_GUN sütunu için 217 farklı veri olduğu Şekil 4.3 teki gibi gözlemlenmiştir.

BAKIYE		NUMBER	0	1.338	67,27	
BIREYSEL_KREDI		VARCHAR2	0	2	0,1006	hayir
EGITIM		VARCHAR2	0	4	0,2011	ortaokul
EV_KREDISI		VARCHAR2	0	2	0,1006	evet
GECEN_GUN		NUMBER	0	217	10,91	

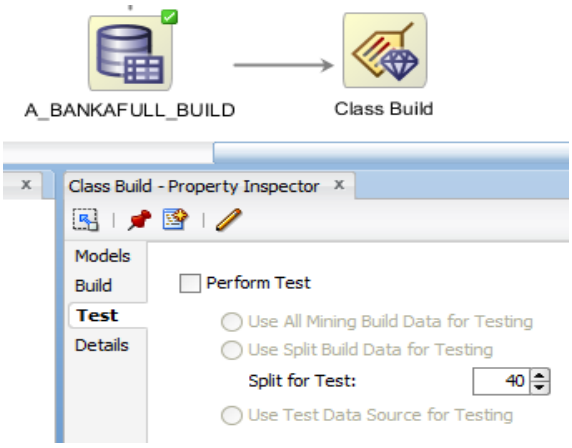
Şekil 4.3: "Explore Data" Sonucu

3. Verilerin analizi yapıldıktan sonra, model oluşturmak üzere "Class Build" düğümü iş akış diyagramına eklenmiştir. "A_BANKAFULL_BUILD" verisi "Class Build" düğümüne bağlandıktan sonra algoritma olarak Naive Bayes algoritması seçilmiş ve "Auto" seçeneği kaldırılmıştır. Veride BAKIYE sütunu için çok fazla sayıda farklı değer bulunduğundan ve ODM'nin her farklı değer için olasılık hesaplamaması istendiğinden "Advanced Settings" bölümünde BAKIYE için "Data Usage" sekmesinde "Auto Preparation" seçilmiştir (Şekil 4.4). Böylece BAKIYE için ODM otomatik olarak gruplar oluşturacak ve bu gruplar üzerinden olasılık değerleri hesaplanabilecektir. Performance Settings sekmesinde "Natural" seçilmiştir.



Şekil 4.4: Model Oluşturma (Aşama 1)

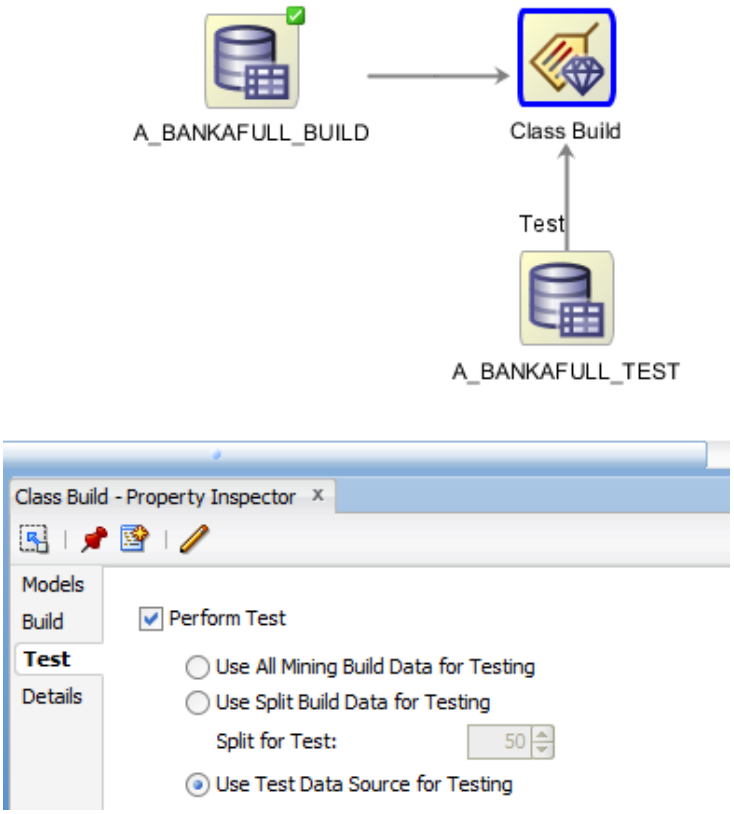
4. "Property Inspector" sekmesinden "Perform Test" seçeneğini kaldırılıp "Class Build" düğümü çalıştırılmıştır (Şekil 4.5).



Şekil 4.5: Model Oluşturma (Aşama 2)

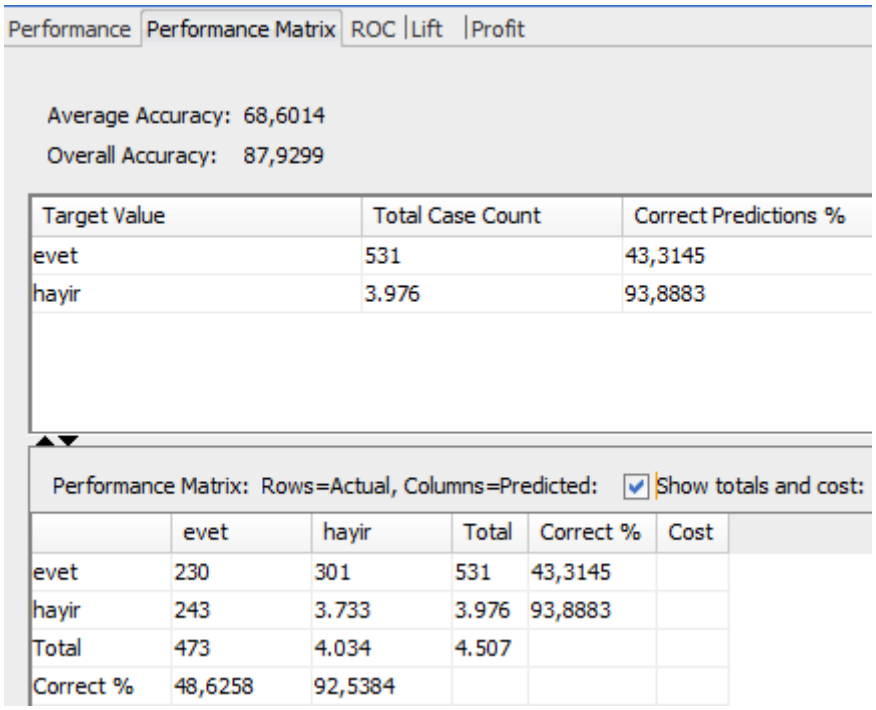
4.4 Modelin Testi

Bir önceki bölümde model oluşturulmuştur. Oluşturulan modelin test verisi üzerinde uygulaması için diyagrama yeni "Data Source" düğümü eklenmiş ve "A_BANKAFULL_TEST" tablosu seçilmiştir. Bu düğüm de "Class Build" düğümüne bağlanmış ve "Property Inspector" bölümünden "Use Test Data Source for Testing" seçeneği seçilip, model tekrar çalıştırılmıştır (Şekil 4.6).



Şekil 4.6: Test Ayarları

Bu adımdan sonra "Class Build" düğümü için "View Test Results" ekranında "Performance Matrix" incelendiğinde modelin doğruluğu %87,9299 olduğu ve 4526 müşteriden bir sonraki kampanya için 303 kişinin aranması gerektiği Şekil 4.7 de görülmüştür.



Şekil 4.7: Modelin Doğruluğu Tablosu

Şekil 4.7 de görüldüğü gibi verideki hayır sayısı, evet sayısına göre çok daha fazladır. Bu yüzden ODM çoğu müşteri için sonucu hayır olarak belirlemekte ve bu nedenle de modelin doğruluğu çok yüksek çıkmaktadır. Öyleki hiç ODM kullanılmadan tüm müşteriler için hayır tahmin edildiğinde, bu tahminin doğruluğu %89,57 çıkmaktadır. Ancak bu durum uygulayıcı için anlamlı bir sonuç ifade etmemektedir. Bu problem için oluşturulan modelin doğruluğununu yanı sıra, asıl amaç aranacak müşteri sayısının yani evet sayısının en üst seviyede tutulmasıdır. Verideki hayır sayısının baskınlığı evet sayısının düşük çıkmasına neden olmuştur. Burada bahsedilen amaca uygun şekilde, olasılığı düşük "hayır" tahminlerinden "evet" olabilecek müşterileri kazanmaya yönelik işlemler yapılmalıdır. Bu işlem yapılırken müşteriye erişim maliyetinin en alt seviyede olmasına dikkat edilir. Bu yüzden modelin evet sayısını arttırabilmek için veri üzerinde "Stratified Sample" işlemi ile eşit dağılımlı (evet ve hayır sayısı eşit) bir veri tablosu tekrar oluşturulmuştur.

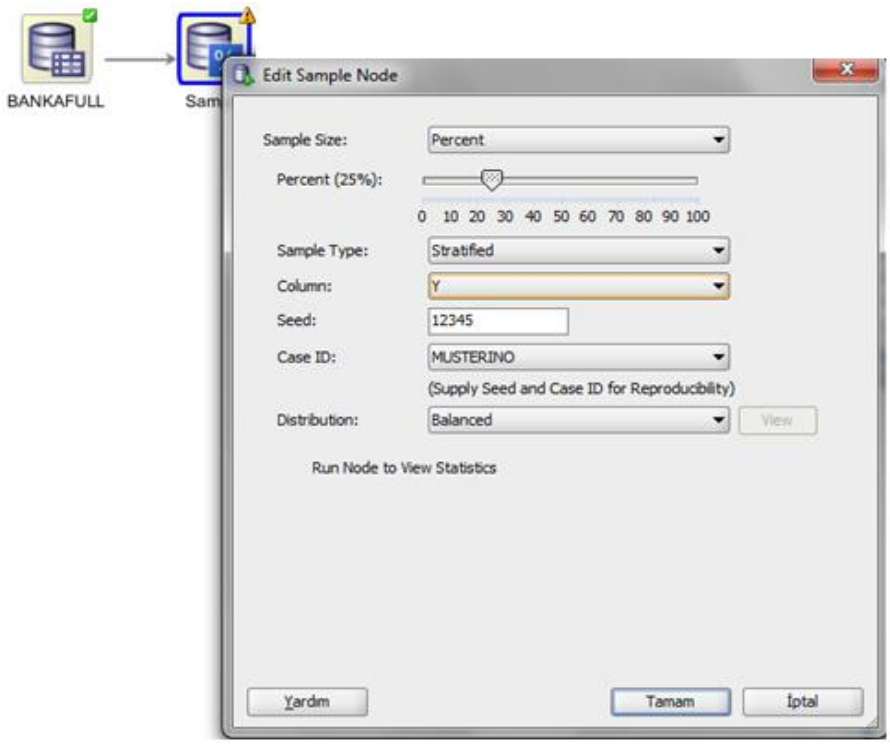
4.5 "Stratified Sample" İşlemi ile Veri Tablosunun Tekrar Oluşturulması

Tablo toplam 45211 veriden oluşmaktadır fakat veri içinde evet sayısı 5289 olup hayır sayısı 39922 dir. Hayır sayısı evet sayısına göre çok fazla olduğundan "stratified sample" işlemi ile evet-hayır sayıları eşitlenmiştir. Böylece yeni verinin

içinde 5289 tanesi evet, 5289 tanesi hayır olacak şekilde toplam 10578 tane veri içeren yeni tablo elde edilmiştir.

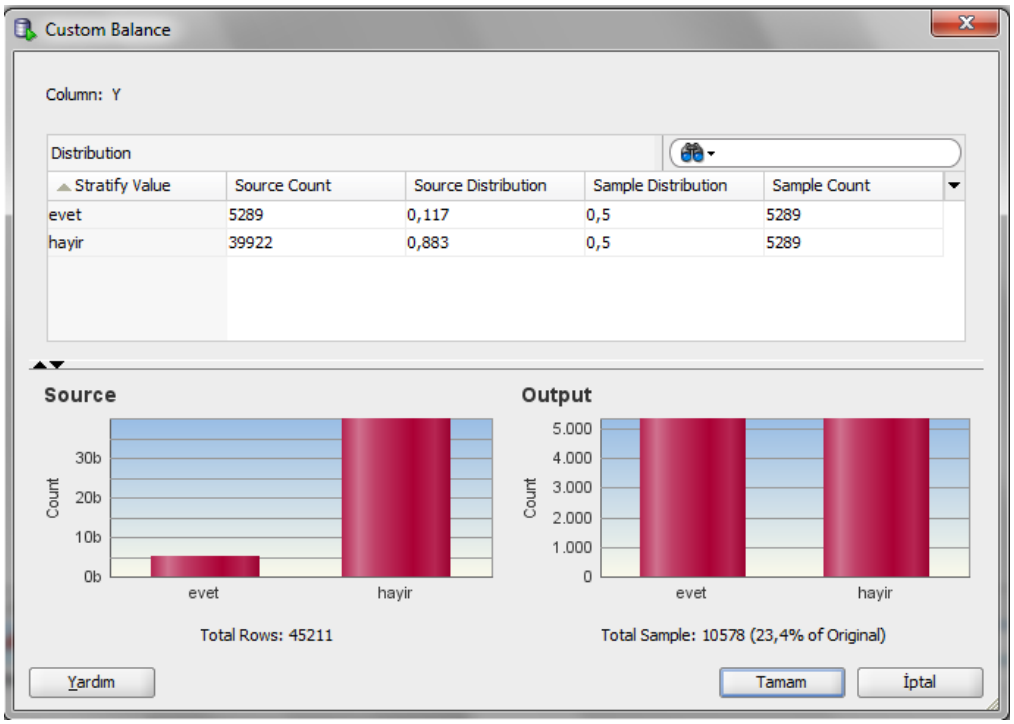
Daha önce 3.2.3 Örnekler başlığı altında Örnek 1 de açıklandığı üzere ilk olarak veri, veri tabanına aktarılmış, "BANKAPROJE" adında bir proje yaratılmış ve projenin içine "ProjeCalismasi" adında bir iş akış diyagramı oluşturulmuştur. Diğer adımlar ise sırasıyla:

1. "Component Palette" kısmından "Data Source" düğümü iş akış diyagramına alındıktan sonra veri olarak daha önceden oluşturulmuş olan "BANKAFULL" tablosu seçilmiştir.
2. Sonra "Component Palette" kısmında "Transforms" sekmesi altında yer alan "Sample" düğümü seçilip iş akış diyagramına eklenmiş ve "BANKAFULL" verisi ile "Sample" düğümü birbirine bağlanmıştır. Daha sonra düğümün ayarları Şekil 4.8 deki gibi seçilerek düğüm çalıştırılmıştır.



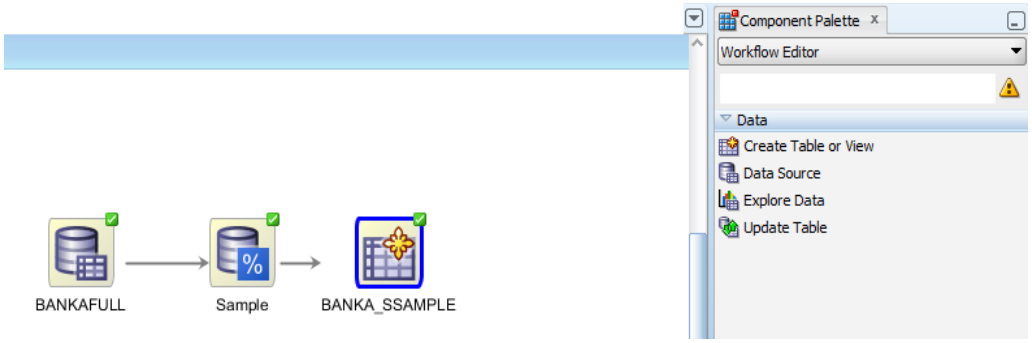
Şekil 4.8: "Sample" düğümü ayarları

3. Özellikleri belirlenip oluşturulan verinin dağılımının gözlemlenmesi için "Sample" düğümü için "Property Inspector" bölümünde "Distrubution" sekmesinin yanında yer alan "View" butonuna basılmıştır (Şekil 4.9).



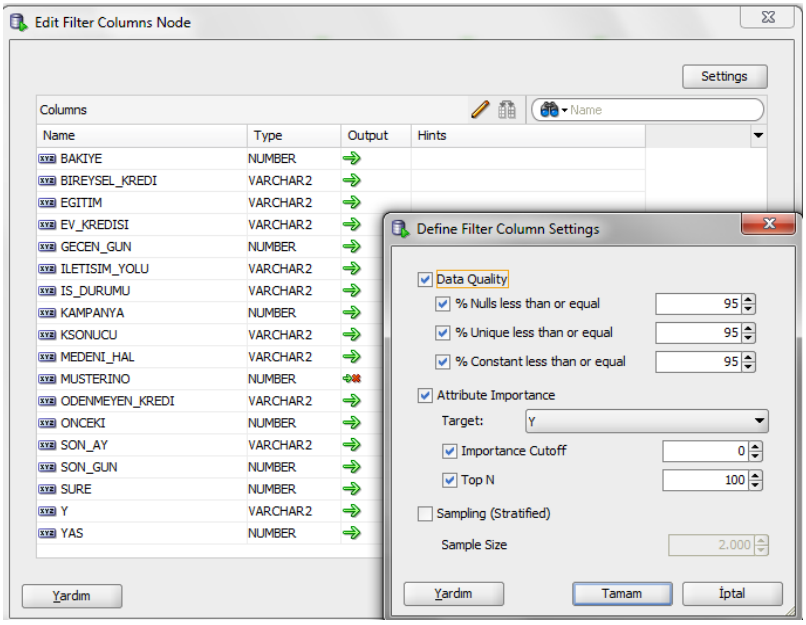
Şekil 4.9: "Stratified Sample" sonucu

4. "Stratified Sample" ile oluşturulan yeni verinin "BANKA_SSAMPLE" adlı bir tablo olarak kaydedilmesi için "Component Palette" kısmında "Data" sekmesi altındaki "Create Table or View" düğümü, iş diyagramına alınmış ve "Sample" düğümü ile bağlanıp çalıştırılmıştır (Şekil 4.10).



Şekil 4.10: Tablo oluşturma

5. Yeni tablo içinde yer alan verilerin olasılıklara olan etkisini görmek için "Attribute Importance" işlemi uygulanmıştır. Bunun için "BANKA_SSAMPLE" verisi "Component Palette" kısmında "Transforms" altında yer alan "Filter Columns" düğümü ile bağlanmış ve ayarları Şekil 4.11 deki gibi düzenlenmiştir.



Şekil 4.11: "Attribute Importance" ayarları

Seçilen ayarlarla düğüm çalıştırılmış, "Filter Columns" düğümü üzerinde "View Data" seçilerek sütunların, olasılıkları nasıl ve ne kadar çok etkileyecekleri incelenmiş ve sonuçları Şekil 4.12 de verilmiştir. Görüldüğü gibi her tahmin edici sütun için "Importance" bölümünde negatif değer olmadığından model inşa ederken hiç bir sütun tablodan çıkarılmadan işlemlere devam edilebilecektir. Ayrıca bu şekilde "SURE" verisinin sonucu en çok etkileyecek sütun olduğuda rankının en yüksek olmasından anlaşılmaktadır.

Attribute Importance | Data | Columns | SQL

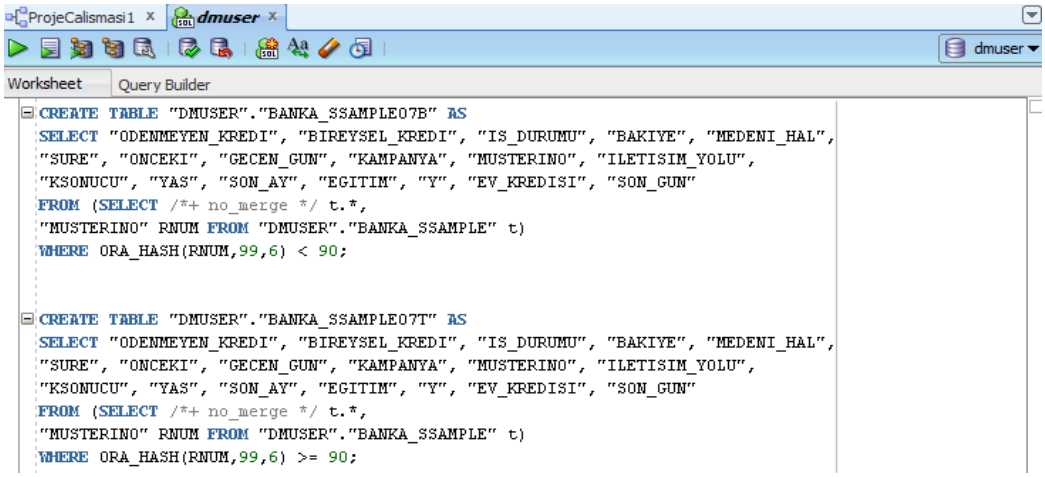
Target: Y

Attribute Ranking

Name	Type	Rank	Importance
SURE	NUMBER	1	0,2327
KSONUCU	VARCHAR2	2	0,0668
SON_AY	VARCHAR2	3	0,0655
GECEN_GUN	NUMBER	4	0,0575
ILETISIM_YOLU	VARCHAR2	5	0,0557
EV_KREDISI	VARCHAR2	6	0,0339
ONCEKI	NUMBER	7	0,0333
YAS	NUMBER	8	0,0315
IS_DURUMU	VARCHAR2	9	0,0253
BAKIYE	NUMBER	10	0,0147
KAMPANYA	NUMBER	11	0,0121
SON_GUN	NUMBER	12	0,0118
EGITIM	VARCHAR2	13	0,0087
BIREYSEL_KREDI	VARCHAR2	14	0,0083
MEDENI_HAL	VARCHAR2	15	0,0057
ODENMEYEN_KREDI	VARCHAR2	16	0,0007

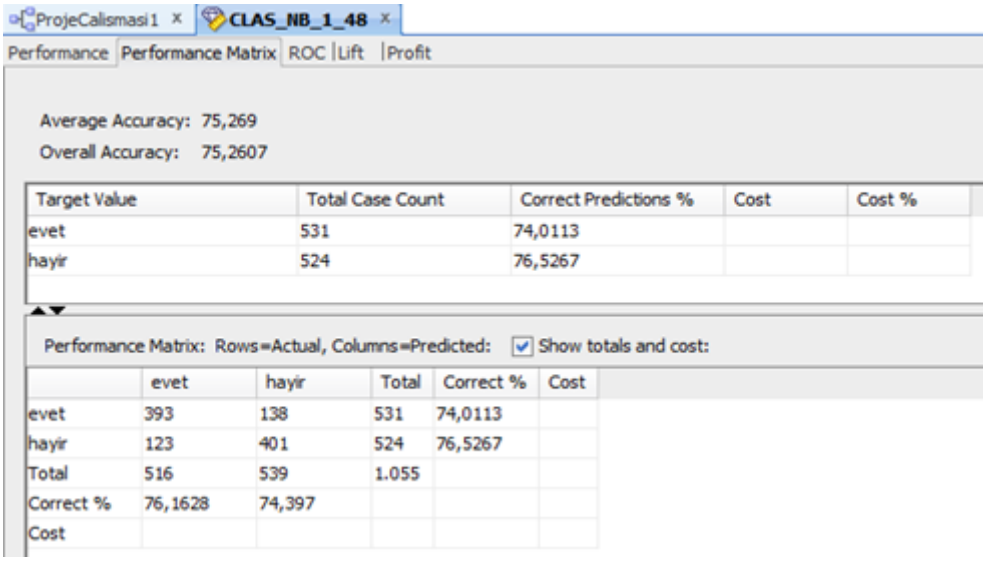
Şekil 4.12: "Attribute Importance"

6. Verinin modelini oluşturup test etmek için veri şekil 4.13 deki ORA_HASH algoritması kullanılan SQL komutuyla %10luk kayıt içeren "BANKA_ SSAMPLE07T" test verisi ve %90lık kayıt içeren "BANKA_SSAMPLE07B" model oluşturma verisi olarak ikiye ayrılmıştır.



Şekil 4.13: Model ve Test verisi oluşturma

Veri tablosu üzerinde yapılan işlemler sonrasında problemin yapısının sınıflandırma modeline uygun olduğu belirlenmiş ve bu bilgiler ile fonksiyon tipi olarak "Classification" çözüm algoritması olarak da "Naive Bayes" seçilerek model oluşturma ve test aşamaları 4.3 ve 4.4 numaralı bölümlerde anlatıldığı gibi gerçekleştirilmiş ve modelin doğruluğu Şekil 4.14 de görüldüğü gibi %75,2607 olarak bulunmuştur.

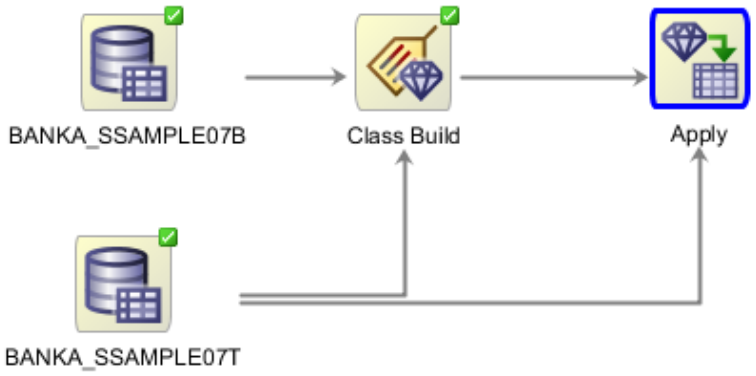


Şekil 4.14: "Performance Matrix" sonucu

4.6 Modelin Uygulanması

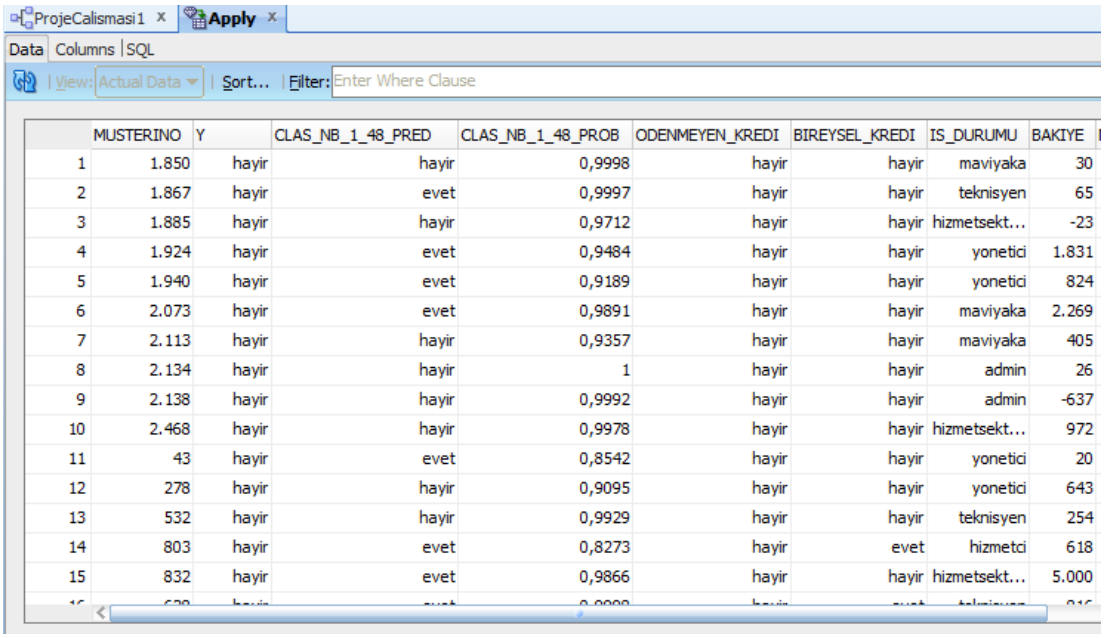
Bu aşamada "BANKA_SSAMPLE07T" test verisine "Apply" işlemi uygulanmış ve test verisi içinde yer alan özelliklere göre her müşteri profili için müşterinin aranıp aranmaması "Evet" ve "Hayır" olarak tahmin edilmiştir. Şekil 4.14 de modelin uygulanması sonucunda 393 tane evet, yani aranacak müşteri sayısı, 401 tane aranmayacak müşteri olduğu sonucu elde edilmiştir.

"Apply" işleminin uygulanması için "Component Palette" de "Evaluate and Apply" sekmesi altında yer alan "Apply" düğümü iş akış diyagramına alındıktan sonra, hem "Class Build" düğümü hem de "BANKA_SSAMPLE07T" verisi ile Şekil 4.15 deki gibi bağlanmıştır.



Şekil 4.15: Modelin Uygulanması

Daha sonra "Apply" düğümünün ayarları düzenlenmiştir. "Data Colums" sekmesinden tüm sütunlar "Selected Attributes" kısmına geçirilip düğüm çalıştırılmıştır. "Apply" düğümünde "View Data" seçilerek verideki gerçek değerleri "Y" sütunu altında, tahmin değerleri ise "CLAS_NB_1_48_PRED" sütunu altında verilmiştir (Şekil 4.16).



ProjeCalismasi1 | Apply

Data | Columns | SQL

View: Actual Data | Sort... | Filter: Enter Where Clause

	MUSTERINO	Y	CLAS_NB_1_48_PRED	CLAS_NB_1_48_PROB	ODENMEYEN_KREDI	BIREYSEL_KREDI	IS_DURUMU	BAKIYE
1	1.850	hayir	hayir	0,9998	hayir	hayir	maviyaka	30
2	1.867	hayir	evet	0,9997	hayir	hayir	teknisyen	65
3	1.885	hayir	hayir	0,9712	hayir	hayir	hizmetsekt...	-23
4	1.924	hayir	evet	0,9484	hayir	hayir	yonetici	1.831
5	1.940	hayir	evet	0,9189	hayir	hayir	yonetici	824
6	2.073	hayir	evet	0,9891	hayir	hayir	maviyaka	2.269
7	2.113	hayir	hayir	0,9357	hayir	hayir	maviyaka	405
8	2.134	hayir	hayir	1	hayir	hayir	admin	26
9	2.138	hayir	hayir	0,9992	hayir	hayir	admin	-637
10	2.468	hayir	hayir	0,9978	hayir	hayir	hizmetsekt...	972
11	43	hayir	evet	0,8542	hayir	hayir	yonetici	20
12	278	hayir	hayir	0,9095	hayir	hayir	yonetici	643
13	532	hayir	hayir	0,9929	hayir	hayir	teknisyen	254
14	803	hayir	evet	0,8273	hayir	evet	hizmetci	618
15	832	hayir	evet	0,9866	hayir	hayir	hizmetsekt...	5.000

Şekil 4.16: Uygulama Sonucu

Şekil 4.16 da ODM ile elde edilen olasılık değerlerine göre sonuçlar yeniden düzenlenirse Tablo 4.2 elde edilir.

Tablo 4.2: Uygulama Sonucunun Olasılık Aralık Değerleri

Olasılık Aralıkları	Evet Yüzdesi	Hayır Yüzdesi	Genel Doğruluk
(0.9,1]	%78,961	%82,6087	%80,74
	304 Müşteri	304 Müşteri	753 Müşteri
(0.8,0.9]	%69,7674	%66,1765	%67,56
	30 Müşteri	45 Müşteri	111 Müşteri
(0.7,0.8]	%72,222	%51,3514	%61,64
	26 Müşteri	19 Müşteri	73 Müşteri
(0.6,0.7]	%70,8333	%41,6667	%53,33
	17 Müşteri	15 Müşteri	60 Müşteri
(0.5,0.6]	%57,1429	%60	%58,62
	16 Müşteri	18 Müşteri	58 Müşteri

ODM ile yapılan bu uygulama sonucunda amaç, bankanın mevcut müşterilerinin kredi satın alıp almama olasılıkları hesaplanarak, uygun müşteri profilini belirlemek ve böylelikle potansiyel müşterilere ulaşmaktır. Diğer bir deyişle Tablo 4.2 de görülen her bir olasılık aralığı için evet yüzdesi, genel ortalama değerleri ve maliyet analizi gözönünde tutularak kredi alması yüksek olasılıklı müşterilere ulaşıp ulaşmamaya karar vermektir.

Bu tablo tahmin edici özellikleri yani bankanın müşterilerine ait yukarıda bahsedilen özellikleri var olduğunda hedef sütunun tamamlanması için müşterilere gelecek kampanyayı tanıtma adına ulaşıp ulaşılmaması, daha doğrusu hangilerine ulaşılması gerektiği konusunda fikir verir. Müşterilerin kayıtlı verileri yardımıyla ODM de oluşturulan bu model yeni müşteriler için kullanıldığında ODM nin müşterinin düzenlenen kampanyaya katılıp katılmayacağı tahmini ve bu tahminin olasılık değeri (0.9-1] arasında olduğunda ODM %80 oranında doğru sonuç vermektedir. Dolayısıyla bu olasılık aralıkları ve karşı gelen genel doğruluk ortalaması hatta daha da iyisi karşı gelen "Evet Yüzdesi" müşteriye ulaşmanın maliyet analizleri ile birlikte değerlendirilerek karar verilebilir. Örneğin, banka görevlisi olasılığı (0.8,0.9] arasında olan bir müşteriyi arandığında %67,56 oranında kampanya için olumlu cevap alacaktır. Müşterinin aranma maliyeti de göz önüne konulduğunda, bankanın bu olasılık aralığındaki müşterileri arayarak kâra geçmesi beklenir. Olasılığı (0.6,0.7] arasında olan müşterilerden %53,33 oranında olumlu yanıt alması mümkündür. Ama burada iletişim maliyeti de hesaplandığında, bankanın bu profildeki müşterilere ulaşması doğru bir yatırım olmayabilir.

Özet olarak, bu veri madenciliği uygulaması ile doğru müşteri profiline ulaşılması, aynı zamanda daha fazla müşteri aranarak zaman ve para kaybının minimuma indirilmesi hedeflenmiştir.

KAYNAKLAR

[1] Yapıcı A., Özel A., Ayça C., 2010. Oracle Data Miner ile Kredi Ödemeleri Üzerine Bir Veri Madenciliği Uygulaması, Bitirme Tezi, İTÜ.

[2] Kırış A., Demiryürek U., 2005. Malzeme Talep ve Sarf Miktarlarının Veri Madenciliği Yöntemleri ile Öngörülmesi.

[3] *http://dilekylmzr.wordpress.com/2010/11/02/oracle-11g-release-2-kurulumu,* Kasım, 2012.

[4] *http://www.oracle.com/webfolder/technetwork/tutorials/obe/db/11g/r2/prod/bidw /datamining /ODM11gR2.html,* Aralık, 2012.

[5] *http://www.oralytics.com/2012/01/odm-11gr2using-different-data-sources.html,* Şubat, 2013.