Pazar Sepeti Analizi için Örneklem Oluşturulması ve Birliktelik Kurallarının Çıkartılması

Sider Hazal Kırtay[1], Nevzat Ekmekçi[1], Tuğba Halıcı[2], Utku Ketenci[2], Mehmet S. Aktaş[1] ve Oya Kalıpsız[1]

[1] Bilgisayar Mühendisliği Bölümü, Elektrik-Elektronik Fakültesi
Yıldız Teknik Üniversitesi, İstanbul
[2] Ar-Ge Merkezi, Cybersoft, İstanbul
hazalkirtay@gmail.com, nvztekmekci@gmail.com, tugba.halici@cs.com.tr, utku.ketenci@cs.com.tr, aktas@yildiz.edu.tr, kalipsiz@yildiz.edu.tr

Özet Bankacılık sektöründe müşteri ihtiyaçlarının doğru, eksiksiz ve hızlı bir şekilde tespit edilmesi ve ürün önerilerinde bulunulması, müşteri memnuniyetinin artırılması için önem arz etmektedir. Her geçen gün artan işlem ve müşteri sayısı nedeniyle analiz yapmak; zaman ve bellek tüketimi açısından maliyetli hale gelmiştir. Bu bildiride, ürün öneri sisteminin performansını artırmak için Pazar Sepeti Analizi sürecine örneklem oluşturma işlemi dahil edilmiştir. Böylece evren yerine; evreni temsil eden ve daha az gözlem sayısına sahip örneklem üzerinde işlem yapılarak, analiz süresi ve bellek kullanımı azaltılabilecektir. Bu kapsamda, örneklem oluşturma yöntemleri, örneklem boyutlarını bulan teknikler, oluşturulan örneklemlerin evrene benzerliğini ölçen testler ve örneklemden çıkartılan birliktelik kuralları incelenmiştir. Müşterilerin ortak satın alma davranışları Apriori algoritması kullanılarak bulunmuştur. Teknikler; eksiksiz kural çıkartma ve harcanan süre kriterlerine göre karşılaştırılmıştır.

Anahtar Kelimeler: Örnekleme, Birliktelik, Veri Madenciliği, Pazar Sepeti Analizi

1 Giriş

Gelişen veri depolama kapasitesiyle birlikte bankaların veritabanları büyüyüp zenginleşmiştir. Çoğu stratejik satış ve pazarlama kararları, bu verilerin işlenmesiyle elde edilmektedir. Örneğin, çapraz satış (cross sell), yukarı satış (up sell) veya risk yönetimi (risk management) gibi stratejiler müşteri verisinin işlenmesi sonucunda oluşturulmaktadır. Artan müşteri sayısı ve buna bağlı artan işlem kapasitesiyle müşteri ihtiyaçlarını hızlı ve doğru bir şekilde tespit ederek çözüm önerileri sunmak daha da zor bir hale gelmiştir. Bu sorunun çözümü için yenilikçi veri madenciliği uygulamaları ve tekniklerine ihtiyaç duyulmaktadır [1].

Pazar sepeti analizi müşterilerin ürün sahiplik verilerinde var olan örüntüyü bulabilmek amacıyla uygulanan veri madenciliği yöntemlerinden biridir. Bu analiz

sayesinde müşterilerin sıklıkla beraber satın aldığı ürünler arasında bir örüntü kurulabilir. Elde edilen örüntü, çapraz satış ve yukarı satış stratejilerinin geliştirilmesinde aktif rol oynamaktadır. Pazar sepeti analizi iki ana süreçten oluşmaktadır. Bunlar sırasıyla kümeleme ve birliktelik süreçleridir. Kümeleme süreci, birbirine benzer müşterilerin kümeler halinde gruplanmasını içermektedir. Böylelikle aynı kategoride incelenmesi gereken müşteriler belirlenmiş olacaktır. Birliktelik sürecindeyse, kümelenmiş ve birbirine benzer özellikteki müşterilerin benzer satın alma davranışı göstereceği varsayımıyla, seçili bir küme üzerinden müşterilerin satın alma davranışlarındaki ortaklıklar tespit edilmektedir.

Bankacılık veritabanlarının çok büyük olması nedeniyle birliktelik işlemi zaman ve bellek tüketimi açısından çok maliyetli bir süreç haline gelmiştir. Zaman ve bellek performansını artırabilmek amacıyla birliktelik öncesine örnekleme süreci dahil edilmesi gerekmektedir. Bu kapsamda evren olarak adlandırılan bütün veri kümesinden daha az gözlem sayısına sahip ve evreni temsil edebilecek bir örneklem oluşturulmaktadır. Elde edilen örneklemin temsil gücünün yüksek olması durumunda bilgi kaybı en aza indirgenmekte ve birliktelik süreci evren yerine örneklem üzerinden gerçekleşmektedir. Böylece daha az veri işlenerek birliktelik kuralları daha hızlı ve daha az bellek harcanarak elde edilmiş olacaktır.

Araştırma konusu banka verisi üzerinde yoğunlaştığından, banka tarafından yapılmış olan müşteri segmentasyonu kümeleme olarak kabul edilmiştir. Segmentasyon sonucunda birbirine benzer müşterilerin oluşturduğu kümeler örnekleme için girdi olarak kullanılmıştır. Bildiride sırasıyla örneklem oluşturma yöntemleri, ideal örneklem boyutunu bulan teknikler, bu tekniklerle üretilen örneklemlerin evreni temsil gücü ve örneklemler üzerinden çıkartılan birliktelik kuralları incelenmiştir. Hem evrenden hem de örneklemden elde edilen birliktelik kuralları örneklem sürecini doğrulamak için kullanılmıştır. Ayrıca, zaman tüketimi açısından ortaya çıkan kazanım hesaplanmıştır.

Bu bildirinin yazım organizasyonu şu şekildedir; Bölüm 2'de; evren ve örneklem üzerinden birliktelik kuralı çıkartmaya yönelik yapılan benzer çalışmalar anlatılacaktır. Bölüm 3'te; birliktelik kuralları elde etmek için gerekli olan parametreler ve Apriori algoritması açıklanacaktır. Bölüm 4; örneklem oluşturmak için gerekli olan parametrelerin, örneklem oluşturma yöntemlerinin ve örneklem büyüklüğünü hesaplamada kullanılan tekniklerin anlatımlarını içermektedir. Bölüm 5'te; evren ve örneklemden elde edilen birliktelik kuralları ve örneklemin evreni temsil etme gücünü gösteren sonuçlar ile zaman tüketimi açısından elde edilen kazanımı gösteren sonuçlar incelenecektir.

2 İlgili Çalışmalar

Birliktelik alanındaki çalışmalar 1990'lı yıllarda ortaya çıkmış ve o dönemden günümüze artarak devam etmiştir [14]. Bu çerçevede ilk önce sıkça beraber bulunan öğe kümeleri çıkartılmakta, daha sonrasında da bu öğe kümelerinden kurallar elde edilmektedir. Birliktelik algoritmaları, elde edilen öğe kümeleri özelliklerine göre sınıflandırılmaktadırlar. Yapılan ilk çalışmalarda geniş öğe kümeleri bulmayı sağlayan Agrawal-Imielinski-Swami (AIS) algoritması kullanılmış,

sonrasında günümüzde de sıkça kullanılan ve büyük veri kümelerini de hızlı işleyebilen Apriori gibi algoritmalar bulunmuştur [2,14].

Birliktelik bulma ve örneklem oluşturma yöntemlerinin beraber kullanılması yeni bir yaklaşım değildir. Birliktelik bulmak amaçlı olarak örneklem oluşturma çalışmaları, evrenin özelliklerini koruyan bir örneklemin oluşturulabileceğini matematiksel olarak ispatlayan makalelerle başlamıştır. Daha sonrasında gerçekleşen çalışmalarda, ideal gözlem sayısını hesaplayan birçok farklı teknik yer almıştır [3,4,6,7].

Örneklem boyutu bulma çalışmalarının başlangıcında örneklenecek veriyle ilgilenilmemiş; hata payı, asgari destek değeri ve asgari güven değeri gibi veriden bağımsız olan parametreler kullanılarak örneklem boyutu hesaplanmaya çalışılmıştır [3]. Devam eden çalışmalarda, verinin özelliklerini de hesaba katan (azami işlem uzunluğu veya veri kümesinin Vapnik–Chervonenkis boyutu gibi değişkenler kullanan) formüller ortaya çıkmıştır [4,6,7].

Bölüm 3 ve 4'te sırasıyla birliktelik bulma ve örneklem oluşturma yöntemleri detaylandırılacak ve çalışmada kullanılan teknikler anlatılacaktır.

3 Birliktelik Bulma

Veri madenciliği alanında birliktelik algoritmaları, gözlemler arasında var olan örüntünün bulunması için kullanılmaktadır [2,8]. Herhangi bir kuruluşun işlem veritabanı ele alındığında; gözlemlerle müşteriler ve örüntü bulunmaya çalışılacak alanlarla satın alınan ürünler arasında bir analoji kurulabilir. Birliktelik algoritmalarının elde ettiği örüntüler işlenerek birliktelik kuralları elde edilmektedir.

Birliktelik kurallarının tanımı şöyle yapılabilir; veritabanında bulunan ürünler kümesinin tüm alt kümelerine öğe kümesi (itemset) ve bir müşterinin beraber satın aldığı ürünlerin oluşturduğu kümeye ise işlem (transaction) diyelim. Herhangi bir öğe kümesinin sahip olduğu destek sayısı (support count) küme elemanlarının veritabanında beraber bulunduğu işlem sayısı olarak tanımlanmaktadır. Öğe kümesinin destek değeri (support) ise destek sayısının veritabanındaki işlem sayısına oranını göstermektedir. Kullanıcı tarafından belirlenen asgari destek değeri (minimum support) koşulunu sağlayan öğe kümesi ise sık rastlanan öğe kümesi (frequent itemset - SRÖK) olarak adlandırılmaktadır. Örneğin, 10 tane işlem barındıran bir veritabanında A ürünü 3 farklı işlemde yer alıyorsa; A ürününün destek sayısı 3, destek değeri ise 0.3' tür. Asgari destek değerinin 0.3'ün altında bir değerle tanımlanması durumunda, A ürünü SRÖK olarak sınıflanır.

Veritabanındaki işlemleri kullanarak SRÖK çıkartan çok sayıda algoritma bulunmaktadır [2,8,9,10]. Evrende var olan bütün öğe kümelerini çıkartması nedeniyle Apriori algoritması tercih edilmiştir. Algoritma, öncelikle veritabanından 1-elemanlı aday öğe kümelerini çıkartmaktadır. Aday öğe kümeleri içinden asgari destek değerini sağlayanlar filtrelenir ve SRÖK olarak kaydedilir. Eleman sayısı artırılarak bir önceki adımda elde edilen SRÖK'den yeni aday öğe kümeler elde edilmektedir. Aday öğe kümeleri her adımda asgari destek değeri sınamasından

geçirilerek k-elemanlı SRÖK üretilemeyene kadar algoritmanın çalışması devam etmektedir. Algoritma 1'de Apriori algoritmasının sözde kodu verilmiştir.

Algoritma 1 Apriori algoritması

Girdi I: ürün kümesi
min_sup: asgari destek değeri
Çıktı SRÖK: sık rastlanan öğe kümeleri

```
k = 0
SRÖK ← ∅
do
    k ← k + 1
    Aday_k ← I kümesinin k-elemanlı alt kümeleri
    SRÖK_k ← ∅
    for all A ∈ Aday_k do
        if supp(A) ≥ min_sup then
            SRÖK_k ← SRÖK_k ∪ A
        end if
    end for
    SRÖK ← SRÖK ∪ SRÖK_k
while SRÖK_k ≠ ∅
```

Veritabanından elde edilen SRÖK'nin elemanları arasında $A \rightarrow B$ formatında birliktelik kuralları (association rule - BK) çıkartılabilmektedir. BK'nin destek değeri $A \cup B$ öğe kümesinin destek değerine eşit olmaktadır. Güven değeri (confidence) ise $A \cup B$ öğe kümesinin destek değerinin A öğe kümesinin destek değerine oranı olarak tanımlanmaktadır. BK'nin kullanıcı tarafından belirlenen asgari güven değeri (minimum confidence) koşulunu da sağlaması gerekmektedir [2].

$A \rightarrow B$ kuralının s destek değerine ve c güven değerine sahip olduğunu varsayarsak, A ve B öğe kümelerinin bütün veritabanında s ihtimalle beraber bulunduğu, ayrıca A öğe kümesine sahip olan müşterilerin c ihtimalle B öğe kümesine de sahip olduğu çıkartımı yapılabilir.

Veritabanındaki bütün BK'nin bulunması için elde edilen her SRÖK'ye kural çıkartma algoritması uygulanmaktadır. Kural çıkartmak için seçilmiş bir SRÖK'nin bütün alt kümeleri arasında $A \rightarrow B$ formatında aday kural kombinasyonları oluşturulmaktadır. Aday kurallar arasından asgari güven değerini sağlayanlar filtrelenir ve BK olarak kaydedilir. Algoritma 2'de kural çıkartma algoritmasının sözde kodu verilmiştir.

4 Örneklem Oluşturma

Örneklem oluşturma, bir veri kümesinin özelliklerini barındıran alt kümesinin oluşturulması işlemidir. Örnekleme sonucunda ortaya çıkan alt kümenin, veri kümesini (evreni) temsil etmesi beklenmektedir.

Geleneksel istatistik yöntemlerinde, iki veri kümesinin benzerliği χ^2 testi veya Kolmogorov-Smirnov (K-S) testi ile ölçülmektedir. χ^2 testi uygulanırken evren özelliklerinin bulunma ihtimalinin %5'ın altında olmaması şartı aranmaktadır.

Algoritma 2 Birliktelik kuralı algoritması

Girdi SRÖK: sık rastlanan öğe kümeleri
min_sup: asgari destek değeri
min_conf: asgari güven değeri
Çıktı BK: birliktelik kuralları

```
BK ← ∅
for all B ∈ SRÖK do
    for all A ⊂ B do
        if conf(A → B) ≥ min_conf and supp(A → B) ≥ min_sup then
            BK ← BK ∪ (A → B)
        end if
    end for
end for
```

Aksi takdirde, testlerde yanlılık oluşabilmektedir [15]. K-S testinde buna benzer bir şart olmamasına karşın, bu testin de 2. tür hata verme ihtimali yüksektir [15]. Oluşturulan örneklemin evrene benzerliğini ölçmek amacıyla bu testlerden faydalanılmıştır. Her iki testin verdiği istatistiklerin p değeri karşılıkları üzerinden bir karşılaştırma yapılabilir. Elde edilen p değeri; 0.05'in üstünde çıkması durumunda "örneklem en az %95 ihtimalle evrene benzerdir" gibi bir çıkartım yapılabilmektedir.

Örneklem oluşturma; örneklem oluşturma yöntemleri ve örneklem boyutu bulma teknikleri olmak üzere iki başlıkta incelenmektedir. Örneklem oluşturma yöntemleri Bölüm 4.1'de, örneklem boyutu bulma teknikleri Bölüm 4.2'de anlatılmıştır.

4.1 Örneklem Oluşturma Yöntemleri

Evrenden örneklem oluştururken, bir çok farklı örneklem oluşturma yöntemi kullanmak mümkündür. Bu yöntemler, evrenden gözlem seçme şekillerine göre sınıflandırılmaktadır. Başlıca örneklem oluşturma yöntemleri şunlardır;

- *Basit rastlantısal örnekleme:* Evrendeki gözlemler herhangi bir rutin izlenmeden rastgele seçilir. Her gözlemin seçilme olasılığı aynıdır.
- *Sistematik örnekleme:* Evrendeki gözlemler numaralandırılır. Evren boyutunun gözlem boyutuna oranı kadar bölüm evrende oluşturulur. Rastgele bir numara seçilir. Her bölümden bu numaradaki gözlem örnekleme dahil edilir.
- *Katmanlı örnekleme:* Evrendeki gözlemlerin gruplara ayrılabileceği durumlarda kullanılır. Grupların/katmanların gözlem sayısının evrendeki toplam gözlem sayısına oranı korunacak şekilde örneklem oluşturulur. Aynı katman içindeki her bir gözlemin seçilme olasılığı aynıdır.
- *Küme örnekleme:* Evrendeki gözlemlerin gruplara ayrılabileceği durumlarda kullanılır. Gruplar belirlendikten sonra basit rastlantısal örneklem yöntemiyle içlerinden seçim yapılır. Seçilen gruplar içindeki bütün gözlemler örnekleme dahil edilir.

– *Çok aşamalı örnekleme:* Evrendeki gözlemlerin gruplara ayrılabileceği durumlarda kullanılır. Gruplar belirlendikten sonra basit rastlantısal örneklem yöntemiyle gruplar seçilir. Küme örneklemeden farklı olarak, gruplar içinden seçilecek olan gözlemler basit rastlantısal örnekleme yöntemiyle belirlenir.

Bahsi geçen yöntemler arasından basit rastlantısal örnekleme yöntemi hızlı çalışmasıyla öne çıkmaktadır. Evrende grup oluşturulmasını ve gözlemler arasında sıralama olmasını gerektiren yöntemler, ön analiz süreci gerektirdiğinden, zaman tüketimleri basit rastlantısal örnekleme yöntemine göre daha fazladır.

4.2 Örneklem Boyutu Bulma Teknikleri

Örneklem oluşturma yöntemlerinde beklenen parametre, oluşturulacak olan örneklemin boyutudur. İdeal örneklem boyutu hesaplanırken, evreni temsil etme gücünü düşürmeyecek bir sayı bulunması gerekmektedir.

Birliktelik algoritmaları kapsamında, oluşturulacak örneklemden evrendeki bütün SRÖK'lerin ve BK'lerin çıkartılması önemlidir. Bu çalışmada, örneklem boyutunu bulmak için geliştirilen tekniklerden, birliktelik algoritmaları için özelleşmiş olanlar incelenmiştir [3,4,6,7]. Örneklem boyutu bulma teknikleri SRÖK ve BK kaybını asgariye indirecek şekilde ikiye ayrılmıştır. Tablo 1'de teknik tipi sütununda SRÖK barındıranlar, SRÖK kaybını; BK barındıranlar ise BK kaybını ortadan kaldırmayı hedeflemektedir.

Tablo 1: Örneklem boyutunu hesaplayan teknikler verilmiştir. Asgari örneklem boyutu; doğruluk ε, aksaklık ihtimali δ, asgari destek değeri Θ, asgari güven değeri γ, evrenin d-indeks değeri v, evrenin azami işlem uzunluğu Δ ve c sabiti cinsinden bulunabilir. Formüllerde geçen η değeri Θ, γ ve ε değişkenlerine; p ise η ve Θ değerlerine bağlı olarak hesaplanmaktadır [5,12,13].

Teknik Adı	Teknik Tipi	Formül
Zaki [3]	SRÖK-mutlak	$\frac{-2\ln(1-\gamma)}{\Theta\delta^2}$
Toivonen [6]	SRÖK-mutlak	$\frac{1}{2\varepsilon^2}\ln\frac{2}{\delta}$
Chakaravarthy [4]	SRÖK-mutlak	$\frac{24}{(1-\varepsilon)\varepsilon^2\Theta}(\Delta+5+\ln\frac{4}{(1-\varepsilon)\Theta\delta})$
Chakaravarthy [4]	BK-mutlak	$\frac{48}{(1-\varepsilon)\varepsilon^2\Theta}(\Delta+5+\ln\frac{5}{(1-\varepsilon)\Theta\delta})$
Riondato [7]	SRÖK-mutlak	$\frac{4c}{\varepsilon^2}(v+\ln\frac{1}{\delta})$
Riondato [7]	SRÖK-bağıl	$\frac{4(2+\varepsilon)c}{\varepsilon^2(2-\varepsilon)\Theta}(v\ln\frac{2+\varepsilon}{\Theta(2-\varepsilon)}+\ln\frac{1}{\delta})$
Riondato [7]	BK-mutlak	$\frac{c}{\eta^2 p}(v\ln\frac{1}{p}+\ln\frac{1}{\delta})$
Riondato [7]	BK-bağıl	$\frac{c}{\eta^2 p}(v\ln\frac{1}{p}+\ln\frac{1}{\delta})$

Aynı parametreler kullanılarak birliktelik algoritmaları koşturulduğunda, örneklemden hesaplanan güven ve destek değerleri, evrenden hesaplanan karşılıkları-

na göre farklı çıkmaktadır. Bu hata payı iki farklı yöntem kullanılarak ölçülmektedir. Mutlak hata payı hesabında, evrenden ve örneklemden bulunan değerlerin farkının mutlak değerine bakılmaktadır. Bağıl hata payı ise, mutlak hata payının evrendeki değere oranı alınarak hesaplanmaktadır. Tablo 1'de teknik tipi sütununda mutlak barındıranlar, mutlak hata payını; bağıl barındıranlar ise bağıl hata payını asgariye indirmeyi hedeflemektedir.

İncelenen bütün teknikler, önerdikleri formüller ve formül tipleriyle birlikte Tablo 1 üzerinde gösterilmiştir. Teknikler sonucunda bulunan değerler örneklem oluşturma için gereken asgari işlem sayısını bulmaktadır. Bulunan değer kadar işlem, belirlenen örneklem oluşturma yöntemiyle evrenden seçilmektedir.

Evrenin karmaşıklığı teorik olarak Vapnik-Chernovenkis (VC) boyutu ile hesaplanmaktadır. Tablo 1'deki formüllerde VC boyutu yerine ona üst sınır olan d-indeks değeri kullanılmaktadır [7,11]. Veritabanındaki işlemlerin, sahip oldukları öğe sayısına göre sıralandıklarını ve işlem sayısı-öğe sayısı eksenli koordinat sistemine yerleştirildiğini varsayarsak, d-indeks değeri en büyük karenin kenar uzunluğuna karşılık gelecektir. Çalışmada önerilen d-indeks algoritması işlemler arasında sıralama koşulu aramamakta ve v'yi 1'den başlatıp artırırarak hesaplamaktadır. Değerin bulunması için veritabanındaki bütün işlemlerin taranması gerekmektedir.

Algoritma 3 İyileştirilen d-indeks hesaplama algoritması

Girdi VT: işlem veritabanı
Çıktı v: d-indeks değeri

```
VT'yi işlemlerin öğe sayısı azalan şekilde sırala
v ← azami işlem uzunluğu
while v ≥ 0 do
    T ← ∅
    for all t ∈ VT'deki en az v öğeye sahip işlemler do
        T ← T ∪ t
        if |T| = v then
            break
        end if
    end for
    v ← v − 1
end while
```

Bankacılık verisi gibi işlem sayısının çok ve her bir işlemdeki öğe sayısının az olduğu durumlarda işlemlerin uzunluğu d-indeks değerini bulmada belirleyici olmaktadır. Önerilen algoritmanın, eldeki veriyle test edildiğinde yavaş çalıştığı gözlenmiştir. Bu nedenle, işlem uzunlukları göz önünde bulundurularak iyileştirilmiş versiyon Algoritma 3'te sözde kod olarak verilmektedir. Veritabanındaki işlemler, öğe sayısı azalacak şekilde sıralanmış ve v değeri azami işlem uzunluğundan başlatılıp azaltılarak hesaplanmıştır. Böylece, bütün işlemlerin taranmasına gerek kalmamıştır.

Tablo 1'de geçen parametrelerin ayrıntıları ve ispatlarına [5,7,12,13] çalışmalarından ulaşılabilir.

5 Test ve Deneysel Değerlendirme

Testler, bankacılık müşterisi ürün sahipliği verisi üzerinde gerçekleştirilmiştir. İstatistiki çalışmalarda yaygın olarak kullanılan R programlama dilinde kod geliştirilmesi yapılmıştır. Testler yapılırken sırasıyla şu aşamalar izlenmiştir:

1. Farklı tekniklerden faydalanılarak örneklem boyutunun bulunması,
2. Hesaplanan boyutun evrenden daha küçük olması durumunda, basit rastlantısal örnekleme yöntemi kullanılarak her bir teknik için 3 ayrı örneklemin oluşturulması,
3. Elde edilen örneklemlerin evreni temsil gücünü karşılaştırmak amacıyla χ^2 ve K-S testlerinin incelenmesi,
4. R dilinin *arules* paketinde var olan Apriori algoritmasından faydalanılarak evren ve örneklem üzerinden SRÖK'nin ve BK'nin elde edilmesi,
5. Sonuçların, evren üzerinden elde edilenlerle karşılaştırılarak destek ve güven değerlerinde oluşan mutlak hatanın hesaplanması,
6. Evrenden BK elde etme süresiyle örneklem oluşturma ve oluşturulan örneklemden BK üretme süresinin karşılaştırılması.

Teorik olarak; tekniklerden elde edilen farklı boyutlardaki örneklemlerin SRÖK ve BK sonuçlarının evrenin sonuçlarıyla uyuşması, temsil gücü ve mutlak hata arasında bir ilişki olması, ayrıca örneklem üzerinden yapılan işlemlerde geçen sürenin ve bellek tüketiminin azalması beklenmektedir.

Test süreçlerini hızlandırmak için bankanın 143 ürünü yerine 10 farklı ürün grubu belirlenmiş ve bu gruplar arasındaki birliktelik araştırılmıştır. Kullanılan banka verisi 1048575 müşteri ve bu müşterilerin 10 farklı ürün grubuna sahiplik durumlarının bulunduğu bir matristir. Satırlar, müşterileri; sütunlar ise ürün gruplarını temsil etmektedir. Müşterinin bir ürüne sahip olması durumunda ilgili satır-sütun kesişimde 1, aksi takdirde 0 bulunmaktadır. Testlerde doğruluk $\varepsilon = 0.04$, aksaklık ihtimali $\delta = 0.07$, asgari destek değeri $\Theta = 0.02$ ve asgari güven değeri $\gamma = \{0.06, 0.1, 0.14\}$ olarak kullanılmıştır. Doğruluk ve aksaklık ihtimali [7]'deki değerler temel alınarak; asgari destek değeri banka verisinin yapısı göz önünde bulundurularak seçilmiştir. Bu üç parametre sabit tutularak farklı asgari güven değerleri [7]'deki gibi test edilmiştir.

Tablo 2 değişen asgari güven değerlerine bağlı olarak değişen örneklem boyutlarını göstermektedir. Toivonen, Chakaravarthy SRÖK-mutlak, Chakaravarthy BK-mutlak, Riondato SRÖK-mutlak ve Riondato SRÖK-bağıl formüllerinde γ'nın parametre olarak kullanılmaması nedeniyle hesaplanan boyutlarda değişim bulunmamaktadır.

Tablo 2 detaylı incelendiğinde; Chakaravarthy SRÖK-mutlak, Chakaravarthy BK-mutlak, Riondato SRÖK-bağıl ve Riondato BK-bağıl tekniklerinden elde edilen boyutların, evrenden (1048575) daha büyük olduğu görülmektedir. Veri

Tablo 2: Değişen asgari güven değerlerine göre hesaplanan örneklem boyutları verilmiştir. Hesaplanan boyutun evrenden daha büyük olduğu teknikler örneklem oluşturma aşamasında kullanılmamıştır.

Teknik Adı	Teknik Tipi	$\gamma = 0.06$	$\gamma = 0.1$	$\gamma = 0.14$
Zaki	SRÖK-mutlak	3867	6585	9426
Toivonen	SRÖK-mutlak	1047	1047	1047
Chakaravarthy	SRÖK-mutlak	14842499	14842499	14842499
Chakaravarthy	BK-mutlak	30033660	30033660	30033660
Riondato	SRÖK-mutlak	9574	9574	9574
Riondato	SRÖK-bağıl	1458404	1458404	1458404
Riondato	BK-mutlak	15057	47005	96859
Riondato	BK-bağıl	5468750	5468750	5468750

kümesinin küçültülmesi amaçlandığından, bu teknikler bir sonraki testlerde incelenmemiştir. Basit raslantısal örnekleme yönteminden kaynaklı hatayı asgariye indirebilmek amacıyla Zaki, Toivonen, Riondato SRÖK-mutlak ve Riondato BK-mutlak tekniklerinin her biri için 3 örneklem oluşturulmuştur.

Tablo 3'te her bir tekniğin χ^2 ve K-S testlerinden hesaplanmış ortalama p değerleri görülmektedir. Evren ve örneklemin benzerlik önem derecesi %95 olarak kabul edildiğinde p değerlerinin 0.05'ten büyük çıkması beklenmektedir. Sonuçlara göre, evren ve elde edilen bütün örneklemler arasında istatiksel olarak benzerlik olduğunu kanıtlamada yeterli olacak değerler elde edilmiştir. Örneklem boyutu değişmeyen Toivonen tekniğinin p değerlerinde kararsızlık göze çarpmaktadır. Bu kararsızlığın, tekniğin verdiği örneklem boyutunun çok küçük olmasından kaynaklandığı düşünülmektedir.

Tablo 3: Değişen asgari güven değerlerine göre χ^2 ve K-S testlerinden hesaplanan p değerleri sunulmuştur. Bütün teknikler evrene benzer çıkmıştır.

Teknik Adı	Teknik Tipi	$\gamma = 0.06$		$\gamma = 0.1$		$\gamma = 0.14$	
		χ^2	K-S	χ^2	K-S	χ^2	K-S
Zaki	SRÖK-mutlak	0.824	0.591	0.630	0.439	0.190	0.382
Toivonen	SRÖK-mutlak	0.379	0.512	0.435	0.395	0.341	0.675
Riondato	SRÖK-mutlak	0.142	0.182	0.595	0.434	0.234	0.081
Riondato	BK-mutlak	0.690	0.618	0.663	0.300	0.385	0.396

Apriori algoritması kullanılarak oluşturulan örneklemlerden SRÖK'ler ve bunlara ait BK'ler bulunmuştur. SRÖK ve BK'lerin benzerliklerini ölçmek için sırasıyla destek ve güven değerleri üzerinden mutlak hata hesaplanmıştır. Zaki ve Toivonen teknikleri $\gamma = 0.1$ değeri için evrende var olan bütün SRÖK'leri ve BK'leri bulmada yetersiz kalmıştır. Kural kaybının istenmemesi nedeniyle, bu iki tekniğin örneklem oluşturmaya elverişli olmadığı görülmüştür ve zaman tüketimi testlerinde incelenmemiştir. Eksik olan değerlerin 0 kabul edilip hata

hesaplamasıyla Tablo 4'teki sonuçlar elde edilmiştir. Beklendiği gibi mutlak destek değeri hatasının yüksek olduğu durumlarda, güven değeri hatası da yüksek çıkmıştır.

Tablo 3 ve Tablo 4 karşılaştırılarak incelendiğinde χ^2 ve K-S testlerinden elde edilen sonuçlar ile destek ve güven hataları arasında bir ilişki bulunamamıştır. Geleneksel istatistik ölçümlerinin birliktelik çıkartmak amacıyla oluşturulan örneklemin temsil gücünü ölçmede yetersiz kaldığı fark edilmiştir.

Tablo 4: Değişen asgari güven değerlerine göre üretilen ortalama destek ve güven mutlak hataları sunulmuştur. BK kaybının yaşandığı teknikler, çalışma süresi açısından test edilmemiştir.

Teknik Adı	Teknik Tipi	$\gamma = 0.06$		$\gamma = 0.1$		$\gamma = 0.14$	
		Destek	Güven	Destek	Güven	Destek	Güven
Zaki	SRÖK-mutlak	0.002	0.009	0.003	0.06	0.002	0.001
Toivonen	SRÖK-mutlak	0.005	0.022	0.006	0.042	0.004	0.001
Riondato	SRÖK-mutlak	0.023	0.008	0.002	0.011	0.002	0.001
Riondato	BK-mutlak	0.011	0.004	0.001	0.001	0.001	0.001

Tablo 5'te BK oluşturmaya kadar geçen süreler evren ve oluşturulan örneklemler için verilmiştir. Evren için, örneklem oluşturmadan BK elde etmeye kadar geçen süre; örneklemler içinse örneklem boyutunu bulma, basit rastlantısal örnekleme yöntemiyle örneklem oluşturma ve BK elde etme için geçen toplam ortalama süre sunulmuştur. Beklendiği gibi her γ değeri için bütün tekniklerin zaman performansları evrene göre daha iyi çıkmıştır. Testlerde 10 tane ürün grubu yerine gerçekteki 143 farklı ürünün kullanılmasıyla kazanımların daha fazla olması beklenmektedir.

Tablo 5: Değişen asgari güven γ değerlerine göre kural çıkartmaya kadar geçen zaman saniye cinsinden verilmiştir. Testler 3.2 GHz i5 işlemciye, 8Gb RAM'e sahip bir makina üzerinde gerçekleştirilmiştir.

Teknik Adı	Teknik Tipi	$\gamma = 0.06$	$\gamma = 0.1$	$\gamma = 0.14$
Evren	–	1.832	1.825	1.87
Riondato	SRÖK-mutlak	0.186	0.193	0.193
Riondato	BK-mutlak	0.193	0.253	0.343

6 Sonuçlar ve Gelecekteki Çalışmalar

Evren üzerinden BK çıkartma işleminin uzun sürmesi nedeniyle, evreni temsil eden daha küçük boyutlu bir örneklemin bulunması ve örneklem üzerinden BK çıkartılması hedeflenmiştir. Bu amaç doğrultusunda, birliktelik bulmak için

özelleşmiş olan ideal örneklem boyutunu veren teknikler araştırılmıştır. Örneklemler, basit rastlantısal örnekleme yöntemi kullanılarak oluşturulmuş ve her bir teknik için 3 farklı örneklem elde edilmiştir. Birden fazla örnek oluşturularak sonuçlarda oluşabilecek olan gürültünün önüne geçilmeye çalışılmıştır. Basit rastlantısal örnekleme yönteminin kullanılmasına ve 3 örneklem oluşturulmasına karar verilirken [7]'deki çalışma temel alınmıştır. Tekniklerin önerdikleri formüller farklı asgari güven değerleriyle test edilmiştir. Testlerde kullanılan değerler test verisinin yapısına ve [7] çalışması temel alınarak oluşturulmuştur. Bu aşamada, göze çarpan ilk değerlendirme, test edilen değerlerde bazı tekniklerin her boyuttaki evrene uygulanabilir olmamasıdır.

Örneklemlerin evrene benzerliği χ^2 testi ve K-S testi ile ölçülmüştür. Her iki test sonucunda elde edilen değerlerin birliktelik çıkartmak amacıyla oluşturulan örneklemlerin temsil gücünü ölçmede yetersiz kaldığı görülmüştür. Söz konusu testlerle, güven ve destek hata değerleri arasında herhangi bir ilişki bulunamamıştır. Testlerin yanlı sonuçlar verme ihtimallerinin olması ve önerilen örneklem boyutlarının ölçüm yapmada yetersiz kalmasının bu sonuçları doğurmada sebep olduğu düşünülmektedir.

Örneklemlerin SRÖK ve BK sonuçları evrenden üretilen sonuçlarla karşılaştırılmıştır. Evrende, en çok rastlanan Mevduat(M) ürünü %94, Kredi Kartı(KK) ürünü %11 ve Taksitli Kredi(TK) ürünüyse %8 oranlarında bulunmaktadırlar. Bu ürünler evren üzerinde %2,3 oranında beraber bulunmaktadır. Bu düşük destek değerine rağmen, güven değeri yüksek 3 farklı kural çıkartılmıştır. $KK, TK \rightarrow M$ kuralı çıkartılmış ve sonuçta KK ile TK alanların %92'sinin M de satın aldığı görülmüştür. Çıkan bir diğer kural $TK, M \rightarrow KK$ ile KK ve M alanların %31'inin TK de satın aldığı ortaya çıkmıştır. Bir diğer kural olan $KK, M \rightarrow TK$ ile de TK ve M alanların %28'inin KK da satın aldığı bulunmuştur. Teknikler içerisinden Zaki ve Toivonen'de bu kurallar çıkartılamamış ve bilgi kaybı yaşanmıştır. Bu nedenle, birliktelik çıkartma amacıyla örneklem oluşturmak için bu tekniklerin kullanılması uygun değildir. Beklendiği gibi örneklem boyutu arttıkça sonuçlarda elde edilen mutlak hata azalmıştır.

Evren üzerinden BK üretme işleminde geçen süreyle; örneklem boyutu bulma, örneklem oluşturma ve örneklem üzerinden BK üretme işlemlerinde geçen toplam süre karşılaştırılmıştır. Her bir tekniğin, evren sonuçlarına göre daha başarılı olduğu görülmüştür. Hesaplanan mutlak hata değerlerine göre Riondato SRÖK-mutlak ve Riondato BK-mutlak teknikleri iyi sonuçlar vermiştir. Daha küçük örneklem boyutu ve daha az zaman tüketimi kriterleri göz önünde bulundurulduğunda Riondato SRÖK-mutlak tekniği öne çıkmaktadır.

Kullanılan bankacılık verisi, müşteriler ve sahip oldukları ürünlerin ürün grupları bilgisini taşımaktadır. Bir başka deyişle, müşterilerin sahip oldukları ürünler yerine bankacılık ürün grupları kullanılmıştır. Bu tercih, veri kümesindeki seyrekliği azaltmak ve testlerin çalışma sürelerini hızlandırmak için yapılmıştır. Bu sayede, evren üzerindeki testlerin dahi 2 saniyeden uzun sürmediği gözlenmiştir. Süre açısından kazanımlar, her ne kadar saniyeler seviyesinde gözükse de daha çok ürün (veya ürün grubu) içeren bir veri kümesiyle yapılacak testlerde daha belirgin kazanımlar gözlenecektir. Gelecek çalışmalarda veri kümesi bu

çerçevede yenilenecek ve diğer örnekleme yöntemleri de uygulanacaktır. Ayrıca, tek bir veri kümesine bağlı olabilecek sonuçlar başka bir veri kümesiyle gerçekleştirilecek testlerle genişletilerek çapraz doğrulamadan geçirilecektir.

Kaynaklar

1. Pulakkazhy, Sreekumar, and R. V. S. Balan. "Data mining in banking and its applications-a review." Journal of Computer Science 9.10 (2013): 1252.
2. Agrawal, Rakesh, and Ramakrishnan Srikant. "Fast algorithms for mining association rules." Proc. 20th int. conf. very large data bases, VLDB. Vol. 1215. 1994.
3. Zaki, Mohammed Javeed, et al. "Evaluation of sampling for data mining of association rules." Research Issues in Data Engineering, 1997. Proceedings. Seventh International Workshop on. IEEE, 1997.
4. Chakaravarthy, Venkatesan T., Vinayaka Pandit, and Yogish Sabharwal. "Analysis of sampling techniques for association rule mining." Proceedings of the 12th international conference on database theory. ACM, 2009.
5. Mannila, Heikki, Hannu Toivonen, and A. Inkeri Verkamo. "Efficient algorithms for discovering association rules." KDD-94: AAAI workshop on Knowledge Discovery in Databases. 1994.
6. Toivonen, Hannu. "Sampling large databases for association rules." VLDB. Vol. 96. 1996.
7. Riondato, Matteo, and Eli Upfal. "Efficient discovery of association rules and frequent itemsets through sampling with tight performance guarantees." Machine Learning and Knowledge Discovery in Databases. Springer Berlin Heidelberg, 2012. 25-41.
8. Zaki, Mohammed Javeed. "Scalable algorithms for association mining." Knowledge and Data Engineering, IEEE Transactions on 12.3 (2000): 372-390.
9. Pei, Jian, et al. "H-Mine: Fast and space-preserving frequent pattern mining in large databases." IIE Transactions 39.6 (2007): 593-605.
10. Borgelt, Christian. "Keeping things simple: Finding frequent item sets by recursive elimination." Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations. ACM, 2005.
11. Vapnik, Vladimir, Esther Levin, and Yann Le Cun. "Measuring the VC-dimension of a learning machine." Neural Computation 6.5 (1994): 851-876.
12. Löffler, Maarten, and Jeff M. Phillips. "Shape fitting on point sets with probability distributions." Algorithms-ESA 2009. Springer Berlin Heidelberg, 2009. 313-324.
13. Har-Peled, Sariel, and Micha Sharir. "Relative (p, ε)-approximations in geometry." Discrete and Computational Geometry 45.3 (2011): 462-496.
14. Agrawal, Rakesh, Tomasz Imieliński, and Arun Swami. "Mining association rules between sets of items in large databases." ACM SIGMOD Record. Vol. 22. No. 2. ACM, 1993.
15. Bircan, H., Y. Karagöz, and Y. Kasapoğlu. "Ki-Kare ve Kolmogorov Smirnov Uygunluk Testlerinin Simülasyon ile Elde Edilen Veriler Üzerinde Karşılaştırılması." Sivas: Cumhuriyet Üniversitesi İktisadi ve İdari Bilimler Dergisi 4.1 (2003).