İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

GRAF BAZLI SORGU SETİ YÖNTEMİ İLE DİFERANSİYEL MAHREMİYETİN SAĞLANMASI

YÜKSEK LİSANS TEZİ

EMİR ESMERDAĞ

Bilişim Uygulamaları Anabilim Dalı

Bilgi Güvenliği Mühendisliği ve Kriptografi Programı

Tez Danışmanı: Doç. Dr. Muhammed Oğuzhan KÜLEKÇİ

ARALIK 2017

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

GRAF BAZLI SORGU SETİ YÖNTEMİ İLE DİFERANSİYEL MAHREMİYETİN SAĞLANMASI

YÜKSEK LİSANS TEZİ

EMİR ESMERDAĞ
(707151008)

Bilişim Uygulamaları Anabilim Dalı

Bilgi Güvenliği Mühendisliği ve Kriptografi Programı

Tez Danışmanı: Doç. Dr. Muhammed Oğuzhan KÜLEKÇİ

ARALIK 2017

İTÜ, Bilişim Enstitüsü'nün 707151008 numaralı Yüksek Lisans Öğrencisi EMİR ESMERDAĞ, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı "GRAF BAZLI SORGU SETİ YÖNTEMİ İLE DİFERANSİYEL MAHREMİYETİN SAĞLANMASI" başlıklı tezini aşağıdaki imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı :	**Doç. Dr. Muhammed Oğuzhan KÜLEKÇİ** İstanbul Teknik Üniversitesi	.....
Jüri Üyeleri :	**Yrd. Doç. Dr. Yusuf YASLAN** İstanbul Teknik Üniversitesi	.....
	Dr. Mahmut Şamil SAĞIROĞLU ERLAB TEKNOLOJİ A.Ş.	.....

Teslim Tarihi : **17 Kasım 2017**
Savunma Tarihi : **13 Aralık 2017**

Annem'e

ÖNSÖZ

Kişisel anlamda gelişimime büyük katkısı olan Babam'a ve ilkokul hocam Sn. Hakan Hiçsönmez'e sonsuz saygı ve hürmetlerimi sunarım.

Lisans öğrenciliğim sırasında hocalığının yanında bir arkadaş olan, bilgisayar mühendisi olmam yolunda bilgi anlamında büyük katkı yapan Sn. Ali İnan hocama, yüksek lisans öğrenimim sırasında bana desteğini esirgemeyen Sn. Muhammed Oğuzhan Külekçi'ye teşekkür ederim. Tez sürecinde verdiği desteklerden ötürü dostum Mustafa Serkan Işık'a çok teşekkür ederim.

Tez çalışmalarına referans olan yayınları beraber yaptığımız Sn. Ali İnan hocama, Sn. Yücel Saygın hocama ve şu anda doktora öğrencisi olan dostum Mehmet Emre Gürsoy'a destekleri ve öğrettikleri için teşekkürü bir borç bilirim.

En büyük teşekkürü Annem, Babam, Abim ve Kız Kardeşime ederim. Beni bugünlere taşıyan ve oğlu olmaktan büyük gurur duyduğum Annem'e özellikle teşekkür ederim.

Aralık 2017

EMİR ESMERDAĞ
(Bilgisayar Mühendisi)

İÇİNDEKİLER

Sayfa

ÇİZELGE LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

GRAF BAZLI SORGU SETİ YÖNTEMİ İLE DİFERANSİYEL MAHREMİYETİN SAĞLANMASI

ÖZET

Bilimin ve teknolojinin gelişmesiyle birlikte dünyamız ve günlük hayatımızın da her bir evresi değişmeye ve gelişmeye devam ediyor. Özellikle yirminci yüzyılda oluşup gelişen bilgisayar ve hesaplamalı bilimler gündelik hayatımızın gelişmesine büyük etki yapmıştır. Bilimin gelişmesi için şunu söyleyebiliriz, bundan bin yıl evvel gözlem ve deneye dayalı olan bilim, son beş yüz yılda teorik bir unsur halini de aldı ve birçok bilimin hem deneysel hem teorik dalları mevcut bulunmaktadır. Ancak son elli yılda gelişen bilişim ve teknolojiyle, birçok disiplin hesaplamalı branşlara da sahip oldu (deneysel fizik, teoriksel fizik, hesaplamalı fizik gibi). Fakat hesaplamalı bilim bilgi yönetiminide içerecek şekilde gelişmektedir ve her gün büyük miktarda veri toplanmaktadır. Sonuç olarak geldiğimiz nokta itibariyle bizler veri çağında yaşıyoruz.

Toplanan büyük miktarda veri arasından gerekli bilgiyi çıkarma işlemine gereksinim duyulmuştur ve bilgi keşif süreçleri için gereken adımları sırasıyla belirtmek gerekirse veri temizleme, veri entegrasyonu, veri seçimi, veri dönüştürme, veri madenciliği, model değerlendirme ve bilgi sunumudur. Veri madenciliği bu süreç içinde, yetenekli metodlar uygulayarak veri modellerini çıkarır. Veri madenciliğinin değişik görevleri mevcuttur ve bu görevleri yüksek seviyede *öngörücü* ve *tanımlayıcı* olarak kategorileştirebiliriz. Sınıflandırma ve Bağlanım görevleri ve bunların algoritmaları, İlişki Kuralı Analizi, Küme Analizi, Metin, Bağlantı ve Kullanım madenciliği mevcuttur.

Veritabanlarına kaydedilen büyük çaplı veri, veri analizcilerine sunulur ve gerekli bilgilerin çıkarılması beklenir. Ancak bu süreçte bir dezavantaj oluşmaktadır. Bu dezavantaj ise veri analizcilerinin veri setinde kaydı bulunan bireylerin hassas verilerine erişebilmesi durumudur. Bunun engellenmesi için Mahremiyeti Koruyan Veri Madenciliği alanı gelişmiştir ve ifşa edilmesi istenmeyen veya izin verilmemiş hassas bilgileri korumak isteyen veri madenciliği alanıdır. Bu alanda yapılan çalışmalar olarak hassas bilgilerin korunması ve mahremiyetin sağlanması için K-Anonimlik, L-Çeşitlilik, T-Yakınlık gibi yöntemler geliştirilmiştir.

2006 yılında ise Dwork, Diferansiyel Mahremiyet'i anlattığı yöntemde etkileşimli olan bir güvenlik mekanizması yöntemi sunmuştur. Dwork bu makalesinde istatistiksel veritabanı güvenliğinin, krpitoloji alanındaki semantik güvenliğin aksine, kişisel verilerin korunmasını garanti etmesinin mümkün olmadığını kanıtlamıştır. Semantik güvenli bir kriptolojik sistemde, görülmeden öğrenilemeyen şifreli mesaj incelenerek açık mesaj hakkında herhangi bilgi elde edinilemez. Ancak istatistiksel veritabanları için benzeri tanımların mahrem bilgileri koruma yöntemleri için mümkün olmadığını ispat edilmiştir. Bunun sebebi de saldırıyı yapacak kişinin yardımcı bir bilgiye sahip olmasıdır. Bu yöntemde bir güvenilir veri toplayıcısının sunduğu arayüz üzerinden kullanıcıların veritabanına sorgular atılması sağlanıp, muhtemelen gürültü eklenmiş cevaplar gösterilir.

Diferansiyel Mahremiyet sadece istatistiksel sorgulara izin verir ve Diferansiyel Mahremiyet'in SQL dilinde uygulandığı bir yöntem geliştirilmiştir. Toplama Fonksiyonlarından bazılarına (COUNT, SUM, MIN ve MAX) uygulanabilir, ancak bu tez çalışmasında sadece COUNT Toplama Fonksiyonu üzerinden ilerlenilmiştir. Buna göre sorguların alanları bulunup grafa döküldükten sonra, NP-Hard bir problem olan sorgu kümesinin hassasiyeti bulunmuştur. Hassasiyet bulunduktan sonra mahremiyet bütçesi olan ε değeri ile bulunan hassasiyet değeri ölçeklenecek şekilde gürültü eklenerek kullanıcıya muhtemelen gürültülü cevapların döndüğü bir yöntem geliştirilmiştir. Bu sayede Diferansiyel Mahremiyet sağlanarak kişilerin hassas verilerinin korunmasına yönelik bir yöntem geliştirilmiştir.

Bu yönteme uygun sorgu setleri oluşturularak veri analizi görevleri uygulanırsa hassasiyetin korunması sağlanmaya çalışılabilir. Bu tez çalışmasında değinilen bilgi kazanımıyla entropi hesaplama ile Öznitelik Seçimi, Ki-kare testi ile Korelasyon Analizi ve Naïve Bayes sınıflandırıcılarla Sınıflandırma işlemlerinde veritabanına atılacak sorguların Graf Bazlı Sorgu Seti Yöntemine uygun olarak atılması durumunda Diferansiyel Mahremiyet sağlanabilir.

DIFFERENTIAL PRIVACY WITH GRAPH BASED QUERY SET

SUMMARY

As the technology and science has been improved, our daily life has changed. Especially, with the invention and improvement of computer at the 20th century, we can say that one of the most important thing that affects human life is computer science.

According to Szalay and Gray, Computational Science is the new branch of most disciplines. However, empirical science was mainly a thousand years ago. After that, in the past five hundred years, theoretical science had been a part for almost every discipline. But now, most disciplines have empirical and theoretical parts. Moreover, in the past fifty years, computational branch has been another part for most disciplines. To give an example and clarify this, we can consider Physics. Physics has different branches; empirical Physics, theoretical Physics and computational Physics. And due to the Computational Science, scientist have to deal with a huge amount of data which is from new scientific instruments, simulations, online data and Internet. So, because of information management by computational science, computer science challenges have been shown up.

There is a popular word which mentions that people are living in the information age. If we understand what data mining is, that word is not correct. Human are living in the data age actually. As it is mentioned, there is a great amount of data that is collected each every day and will be. Some people take data mining as a synonym for knowledge discovery from data, whereas some take it as just a step in the knowledge discovery process. These steps are data cleaning, data integration, data selection, data transformation, data mining, pattern evaluation and knowledge presentation. Data cleaning, data integration, data selection and data transformation are considered as preprocessing for data mining. And at the mining step some intelligent methods are applied to find patterns. Then with evaluation where interesting patterns due to interesting measures are represented. These are the steps to turn data into knowledge.

It is a common approach that data are published for analysis. However, there is a privacy risks behind of sharing data. This privacy risk and possible disadvantage is the disclosure of sensitive information of individual. There is an area to prevent this risk which is called Privacy Preserving Data Mining. According to the description of Evfimievski and Grandison, this area tries to safeguard sensitive information from unsolicited or unsanctioned disclosure.

To mention some studies of the Privacy Preserving Data Mining area, K-Anonymity, L-Diversity, T-Closeness has been published. K-Anonymity is a method which considers the quasi-identifiers, L-Diversity is a technique beyond K-Anonymity which considers diversity of sensitive data. And T-Closeness is another method beyond both K-Anonymity and L-Diversity. With T-Closeness method, it is aimed that the distribution of sensitive data in a group should be close to all data.

Differential Privacy is a protection mechanism by Dwork. According to Dwork, there is no absolute guarantee by statistical database security, where semantic security in cryptography can guarantee to individuals. From a semantically secure cryptosystem, we can not gain any information about text-plain by cipher-text which can not be learned without seeing it. But Dwork proved that, same definition is not possible due to the auxiliary information. For example, supposing an attacker who knows an individual's height is 2 inches shorter than the average height of women in a country. And the database gives the information of the average height of women in the individuals' country. Then the attacker can exactly know the height of the individual. So there is always some risk for sensitive data for any statistical database.

Moreover, there is two privacy mechanism models. One is non-interactive and the other one is interactive. With the non-interactive model, the data which has sensitive information is sanitized before it is published and shared. K-Anonymity is an example for this type of model. On the other hand, with interactive model, a trusted data collector provides an interface so that the interface users can pose queries and get the possible perturbed answers. Differential privacy is an example to interactive models.

In this study, an approach to achieve the differential privacy is studied and explained. This model is fit for SQL which is very common in information technologies. With the approach and method only some statistical queries are considered to be answered with a provided interface. These statistical queries can have some aggregate functions which are COUNT, SUM, MIN and MAX. However, as an aggregate function COUNT is mainly considered and explained in this study.

As the step for Differential Privacy, computation the sensitivity of a query set is NP-hard. But in the study, an approach to calculate the sensitivity of the query set is explained. So the solution is that building region-intersection graph for non ignored queries. After the intersections are measured, a graph is generated where each every node represents a query, and edges between nodes represents intersection of regions. Then it is showed that computation of the sensitivity of the query set is equivalent to bounding the sensitivity from above.

After the sensitivity of the query set is found, Laplace noise is added to the each every non ignored query. To add the Laplace noise, there is two magnitude; the sensitivity of the query set and privacy budget ε. Then the users of the interface get the possibly perturbed answers and with the model Differential Privacy is provided for only statistical queries which fits the model.

Moreover, it is possible to use this model for data analysis techniques. The model fits for some data analysis techniques because in order to apply the technique, the queries can be generated based on the model in this study. Although there is a lot of data analysis models fits with the model, there is some implemented and mentioned ones; Feature Selection with entropy, Correlation Analysis by chi-square test and Classification with Naïve Bayes Classifiers. All of these three data analysis techniques can be applied with SQL queries which fits the model explained. So, it is possible to analyze the data while protecting the sensitive information of individuals.

Feature Selection is the process to calculate the top-k attributes with the lowest entropy. So that, the lowest entropy offers the highest information gain. The top-k attributes can be used to generate decision trees. Correlation Analysis is for analysis the correlation between the attributes of a table. In the study, Chi-square test is used to calculate the

correlation between attributes. Naïve Bayesian Classifiers which are statistical, tries to find out the probabilities for a given tuple the belonging class. For example, according to a training data set, the belonging class probabilities of a tuple are found and decision is made.

1. GİRİŞ

Teknoloji her geçen gün gelişmektedir. Özellikle 20. yüzyılda gelişmeye başlayan ve yakın gelecekte en çok gelişmesi beklenen bilgisayar bilimi ve teknolojisi sayesinde, bilim muazzam derecede hayatımıza etki etmektedir. Geldiğimiz nokta itibariyle bilgisayar kullanmayan birey, yazılım ve veritabanı teknolojilerini kullanmayan şirket ve kuruluş neredeyse kalmamış bulunmaktadır.

Devlet dairelerinde yaptığımız işlemlerden tutun, internet üzerinden yaptğımız işlemlerde, taşınabilir teknolojik aletlerle yaptığımız işlemlerde yani kısacası hayatımızın her anında teknolojiyi kullanmaktayız. Bu kullandığımız bilgisayar ve yazılım teknolojisi ürünleri (web siteleri, mobil uygulamalar, bankacılık işlemleri, hesaplamalı bilimler için uygulanan yazılımlar, e-devlet işlemleri vs.) üzerinden yapılan işlemler veya gerekli veriler veritabanlarına kaydedilir. Veritabanlarına kaydedilen veriler gerekliyse daha sonra işleme konulabilir, değiştirilebilir, silinebilir veya güncellenebilir. Veya bir başka işlem süreçleri için veritabanlarına kaydedilen veriler kullanılabilir.

1.1 Tezin Amacı

Veritabanlarına kaydedilen veriler çoğu zaman veri analizcilerine sunulur. Veri analizcileri, gerekli uygun yöntemleri kullanarak verileri işler. Ancak burada dikkat edilmesi gereken bir durum mevcuttur ve bu durum ise veri analizcilerinin, veritabanında kaydı bulunan bir bireyin hassas verilerini öğrenememesi gerektiğidir. Bir kişinin hassas verilerinin öğrenilmesi kişinin izni olmadığı durumlarda illegal olabilir. Bu yüzden veri analizcilerine sunulan verinin veya sunma yönteminin kişilerin hassas verilerini açığa çıkarmayacak şekilde olması gerekmektedir. Bu tez çalışmamda da veri analizi sırasında hassas verilerin açığa çıkmasını engellemeye çalışan bir yöntemi anlatmaya çalışacağım. İlerideki bölümlerde, Dwork tarafından açıklanan Diferansiyel Mahremiyet [1] yöntemiyle mahremiyetin nasıl sağlanabileceğine yönelik bir çalışma anlatılıyor.

2. VERİ MADENCİLİĞİ

2.1 Veri Çağına Nasıl Geldik?

Her gün muazzam miktarda verilerin toplandığı bir dünyada yaşıyoruz. Bu kadar büyük çapta verilerin toplandığı dünyaya bilimsel yaklaşım ve disiplinlerdeki değişim ile ulaştık. Bunu biraz açıklamak gerekirse [2] : bundan bin yıl evveline kadar bilim öncelikli olarak gözlem ve deneye dayalı olarak ilerliyordu. Ancak tüm disiplinler son beş yüz yılda teorik bir unsur da oldular. Teorik modeller sayesinde genelde deneyler gerçekleştirilir ve anlayışımız genelleştirilir. Bugün baktığımızda bir çok disiplinin hem deneysel hemde teoriksel dallara sahip olduğu görülür. Son elli yılda bir çok disiplin hesaplamaya dayalı bir branşa da sahip oldu (deneysel fizik, teorik fizik, hesaplamalı fizik vb.). Hesaplamalı branş ile karmaşık matematiksel modeller simüle edilir. Lakin hesaplamalı bilim, bilgi yönetimini de içerecek şekilde gelişmektedir ve bilim insanları her gün büyük miktarda yeni bilimsel enstrümanlardan gelen veriler, simülasyonla gelen veriler, petabaytlarca online veriler ve internet üzerindeki arşivlerden gelen verilerle karşılaşmaktadır.

Bir disipline örnek olarak astronomi bilimini ele alırsak [2], Astronomi binlerce yıl önce öncelikli olarak deneysel dalı olan ve sadece birkaç teorik modeli olan bir bilimdi. Teorik Astronomi Kepler'le başladı ve bugün gözlemler ile eş değer. Astronomi, başlarda hesaplamalı teknikler benimsenerek yıldızlar, galaktik oluşum ve gök mekaniğinin modellenmesiydi. Bugün ise simülasyon bu alandaki önemli parçalardan biridir ve bu sayede yeni bilimler üretiliyor ve mevcut teoriler sağlamlaştırılıyor. Eski günlerde astronomlar dikkatlice her bir fotoğraf plakalarını analiz edebilirlerdi.

Sonuç olarak geldiğimiz nokta ve yaşadığımız çağ gereği bizler veri çağında bulunmaktayız [3].

2.2 Veri Madenciliği Nedir?

Günümüzde bilim insanları tarafından popüler bir araştırma alanı olan Veri Madenciliği, veri çağında yaşayan bizler için veriden bilgi edinme gereksiniminden

dolayı doğmuştur [3]. Burada vurgulanması gereken, her ne kadar popüler bir söz olan *"Bilgi Çağında Yaşıyoruz."* sözü söylense de aslında bizler veri çağında yaşıyoruz [3]. İnternet ortamına, World Wide Web ve çeşitli veri kayıt cihazlarına iş hayatından, toplumdan, bilim ve mühendislikten, tıp biliminden ve günlük hayatımızda bulunan bir çok unsurdan çok miktarda veri akışı olmaktadır [3].

Tabi ki büyük miktarda veri akışının sonucu olarak, bu büyük miktarda verileri kaydedebilecek güçlü veri depolama cihazlarının gelişimi hızlanmıştır [3]. İş dünyasından gelen satış işlemlerini, ticari stok kayıtlarını, ürün tanımlarını, satış promosyonlarını, şirket profillerini ve perfonmasını, ve müşteri geri bildilerimlerini içeren devase veri akışı olmaktadır [3]. Örneğin Wal-Mart gibi büyük mağazalar, tüm dünyadaki binlerce mağazasından haftalık yüz milyonlarca işlemi idare etmektedir [3].

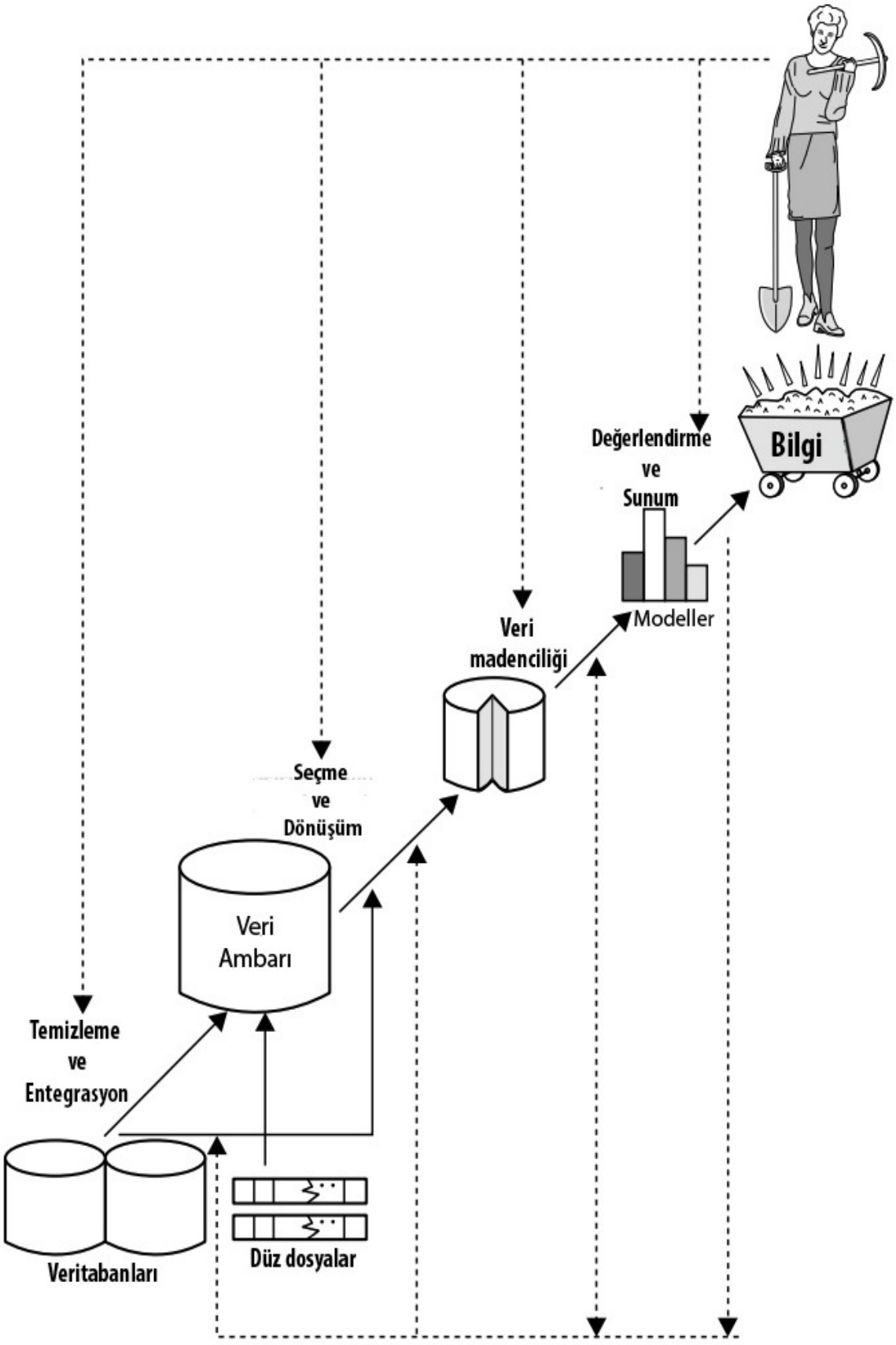
Her gün hayatımızın değişik unsurlarından binlerce veri zamanımızı gerçekten veri çağı yapmaktadır [3]. Bu büyük miktarda veri yığınından organize edilmiş bilgiyi elde etmek içi otomatik olarak güçlü ve bir çok iş yapabilen çok yönlü araçlara ihtiyaç duyulmuştur ve bu gereklilik veri madenciliğini doğurmuştur [3]. Yani veri madenciliği büyük miktarda toplanmış veriyi bilgiye çevirir [3].

2.3 Veriden Bilgi Madenciliği

Veri Madenciliği, Bilgi Keşfi'ndeki süreçlerden biridir [4].

Gerekli bilgiyi elde etmek için veriden bilginin keşif süreci için bazı adımlar mevcuttur [4]. Şekil 2.1'de bu süreçler gösterilmiştir [4]. Her ne kadar burada kısaca tanımları yapılsa da her biri kendi içinde detaylı süreçler içeren bölümlerdir. Büyük çaplı veriden bilgi keşfi için gerekli süreçleri sıralamak gerekirse [4]:

1. Veri Temizleme: Gürültülü ve tutarsız verileri veri tabanından çıkarma,
2. Veri Entegrasyonu: Farklı ortamlardan gelen verileri birleştirme,
3. Veri Seçimi: Analiz işi için gerekli olan verileri seçmek,
4. Veri Dönüştürme: Özetleme veya toplama işlemleri yapılarak veriyi uygun hale getirme,
5. Veri Madenciliği: Yetenekli metodlar uygulanarak veri modellerini çıkarmak,



Şekil 2.1 : Bilgi keşfinde bir adım olarak Veri madenciliği [4].

6. Model Değerlendirme: Gerçekten işe yarayacak ilginç modelleri belirlemek,

7. Bilgi Sunumu: Görselleştirme ve bilgi sunumu teknikleriyle çıkarılan bilgiyi sunmak.

3. VERİ MADENCİLİĞİ GÖREVLERİ

Veri madenciliğini anlamanın en iyi yollarından biri görev çeşitlerini anlamaktır [5]. Yüksek bir seviyede Veri madenciliği görevlerini *öngörücü* ve *tanımlayıcı* görevler olarak kategorileştirebiliriz [5]:

- *öngörücü* : bu tip görevlerde, mevcut olan diğer bilgileri temel alan bir değişkenin değerini tahmin etmeye izin verilir.
- *tanımlayıcı* : bu tip görevlerde ise veriyi bir şekilde özetlemeye çalışılır.

3.1 Öngörücü Görevler: Sınıflandırma ve Bağlanım

Sınıflandırma ve Bağlanım görevleri, en çok karşılaşılan veri madenciliği görevleridir [5]. Bu görevler [5], bir objeyi daha önceden belirlenmiş bir sınıfa (sınıflandırma) veya bir nümerik bir değere (bağlanım) eşleştirmeyi kapsar. Sınıflandırma görevlerinde hedef değişken birbirinden farklı ayrık değerlere sahip, 'high' veya 'low' gibi, bağlanımda ise hedef değişken sürekli bir değere sahip olur [5].

Çizelge 3.1 : Örnek Araç Kredisi Geri Ödememe Verisi [5].

Age	Income	Student	Credit Rating	Default
Youth	Medium	Yes	Fair	No
Youth	Low	Yes	Fair	No
Senior	Low	No	Excellent	No
Senior	Medium	No	Excellent	No
Senior	High	No	Poor	Yes
Senior	Medium	No	Poor	Yes
Senior	Low	Yes	Fair	No
Middle Age	Low	No	Fair	Yes
Middle Age	Medium	Yes	Fair	No
Middle Age	Low	No	Fair	Yes

Çizelge 3.1'de gösterilen veri, müşterileri *Default* özniteliğine göre sınıflandırmaya yarayan *öngörücü* modeller üretmek için kullanılabilir [5]. Yani müşterilerin *Age,*

Income, Student, Credit Rating özniteliklerine göre kusurlu olup olmadığı '*Yes*' veya '*No*' diye *Default* özniteliğine göre sınıflandırılabilir.

1) If credit-rating = "Poor" -> Default = "Yes"

2) If Age = "Middle Aged" and Income = "Low" -> Default = "Yes"

3) -> Default = "No"

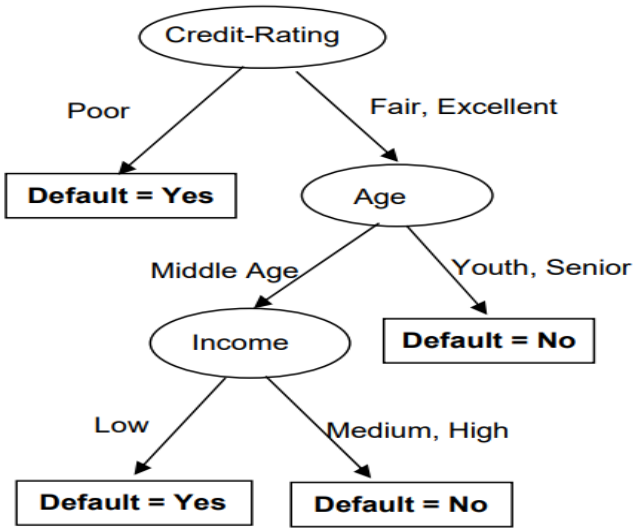
Şekil 3.1 : Araç Kredisi Geri Ödememeleri Tahmin Etmek İçin Kural Kümesi [5].

Veri madenciliğinde *öngörücü* model eğitim setlerinden üretilir [5]. Bu yöntem ile, tüm eğitim örnekleri bir denetleyici tarafından sağlanarak, denetlenen öğrenme yapılır. Örnek vermek gerekirse Şekil 3.1'deki görselde, Çizelge 3.1'den yani eğitim setinden kural üretilmiştir.

3.1.1 Öngörücü veri madenciliği algoritmaları

3.1.1.1 Karar ağaçları

Karar ağacı algoritmaları çok popülerdir ve bunun sebebi anlaşılmasının çok kolay olmasıdır [5]. Çok hızlı şekilde üretilebilip yeni örnekler eklenebilse de, birim zamanda sadece tek bir özniteliği dikkate alabilmesi bir dezavantajıdır [5]. Şekil 3.2'de oluşturulan bir karar ağacı modeli gösterilmiştir [5].



Şekil 3.2 : Araç Kredisi Verisinden Elde Edilen Karar Ağacı Modeli [5].

3.1.1.2 Kural bazlı sınıflandırıcılar

Kural bazlı sınıflandırıcılar sınıflandırma kuralları üretir [5]. Şekil 3.1'de verilen kural seti buna bir örnektir [5].

3.1.1.3 Yapay sinir ağları

Beynin bazı fonksiyonlarından esinlenerek, sınıflandırma ve bağlanım yapmak amacıyla kullanılır [5].

3.1.1.4 En yakın komşu

Günlük hayatta sürekli kullandığımız basit bir örnekle açıklamak gerekirse [5], emlakçılar satacakları yeni evin fiyatını, bölgede satılan benzer evleri baz alarak benimser.

3.1.1.5 Naïve bayes sınıflandırıcılar

Naïve Bayes sınıflandırıcılar olasılıksal sınıflandırıcıdırlar [5]. Tezle ilgili olduğundan, Naïve Bayes sınıflandırıcılar Bölüm 7.3.1 detaylı şekilde açıklanmıştır.

3.2 İlişki Kuralı Analizi

Çizelge 3.2 : Bakkal Dükkanından Market Sepet Verisi [5].

Transaction ID	Items
1	Ketchup, Hamburgers, Soda
2	Cereal, Milk, Diapers, Bread
3	Hot Dogs, Ketchup, Soda, Milk
4	Greetig Card, Cake, Soda
5	Greeting Card, Cake, Milk, Cereal

Çizelge 3.2'de gösterilen market sepet verisinde, her bir ödemede alınan ürünler listelenmiştir ve aynı ödemede beraber alınan ürünlerin bulunması hedeflenmiştir [5]. Bu sebeple, Çizelge 3.2 üzerinden, gerekli veri madenciliği metodları kullanılarak ilişki kuralları belirlenebilir.

3.2.1 İlişki kurallarının belirlenmesi

İlişki kurallarını belirlemek için izlenecek yol şu şekildedir [5]; Kullanıcı tarafından belirlenen *minsup* değeri ile, bir ilişkideki tüm öğelerin beraber bulunduğu ödemelerin sayısının, tüm ödemelerin sayısına oranı ile minimum destek seviyesi belirlenir. Yine kullanıcı tarafından belirlenen *minconf* değeri ile de, ilişkideki tüm öğelerin geçtiği ödemelerin sayısının, sol tarafta bulunan öğelerin geçtiği ödemelerin sayısına oranı bulunarak minimum güven seviyesi bulunur. İlişkilerin kural haline gelebilmesi için, hesaplamalarda bulunan değerlerin bunlardan az olmaması gerekmektedir.

Örneğin [5], *minsup* değerinin 0.4 ve *minconf* değerinin 0.75 olarak belirlendiğini farz ederek, Çizelge 3.2'e bakarak [5], $\{\text{Ketchup}\} \rightarrow \{\text{Soda}\}$ ilişkisinin bir kural olup olamayacağını inceleyelim. Ketchup ve Soda öğelerinin beraber bulundukları ödemelerin sayısının, tüm ödemelerin sayısına oranı $2/5$ yani 0.4'tür ve *minsup* değerine denktir. Ketchup ve Soda öğelerinin beraber bulunduğu ödemelerin sayısının, sadece Ketchup öğesinin bulunduğu ödemelerin sayısına oranı $2/2$ yani 1'dir ve *minconf* değerinden fazladır. Dolayısıyla $\{\text{Ketchup}\} \rightarrow \{\text{Soda}\}$ ilişkisi bir kural ilişkisi olabilir. Çizelge 3.3'de gösterilen ilişki kuralları [5], *minsup* değerinin 0.4 ve *minconf* değerinin 0.75 olarak belirlendiğini farz ederek, Çizelge 3.2'e bakarak [5] elde edilmiştir.

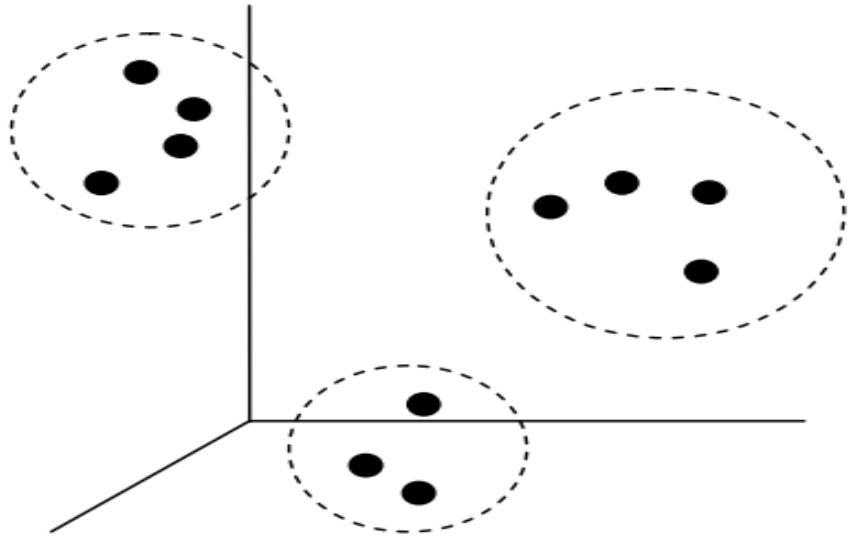
Çizelge 3.3 : Market Sepet Verisinden Üretilen İlişki Kuralları (*minsup* = 0.4, *minconf* = 0.75) [5].

Association Rule	Support	Confidence
Ketchup -> Soda	0.4	1.0
Cereal -> Milk	0.4	1.0
Greeting Card -> Cake	0.4	1.0
Cake -> Greeting Card	0.4	1.0

3.3 Küme Analizi

Küme analizi otomatik olarak verileri karakteristiklerine göre parçalayıp anlamlı gruplara ayırır [5]. Birbirine benzeyen öğeler aynı gruba, birbirlerine benzemeyenler ise farklı gruba yerleştirilir. Kümeleme bir denetlenmemiş, yani eğitim setinden

cevapların elde edilmediği bir görevdir. Şekil 3.3'de bir kümeleme örneği gösterilmiştir [5].



Şekil 3.3 : On bir Örnek Üç Tane Kümeye Yerleştirilmiş [5].

3.4 Metin, Bağlantı ve Kullanım Madenciliği

Bu bölümde etkin bir biçimde geleneksel ilişkisel veritabanlarına kaydedilemeyen, yapılandırılmamış, yarı yapılandırılmış, sayısal olmayan veriler üzerinde madencilikten bahsedilecek [5].

- Metin madenciliği: Metin madenciliğinin temel birimi dökümandır ve bir döküman isteğe bağlı büyük bir sözlükten keyfi sayıda terim içerebilir [5].
- Metin gösterimi: Bir döküman, bir sözcükler çantası gibi muamele görüp, önemine tekabül eden ağırlığıyla, bazı terimler ihmal edilir, bazıları saklanır [5].
- Metin sınıflandırması ve kümelemesi: Metin sınıflandırmasında Naïve Bayes algoritması ve destek vektör makineleri yaygın olarak kullanılır [5].
- Bağlantı madenciliği: Birçok çeşit verinin birbiri arasında bağlantılanarak karakterize edilmesidir [5].
- İçerik madenciliği: Web sayfalarında madencilik için muazzam büyüklükte olgunlaşmış veri bulunmaktadır [5].
- Web kullanım madenciliği: Kullanıcıların web siteleri üzerindeki hareketlerinin davranışsal modelleri keşfedilip analiz edilebilir [5].

4. MAHREMİYETİ KORUYAN VERİ MADENCİLİĞİ

4.1 Mahremiyeti Koruyan Veri Madenciliği Nedir?

Veri Madenciliği, gerekli bilgiyi elde etmek için veriden bilginin keşif süreci için gerekli adımlardan biridir [4]. Ancak veri madenciliği yapılarak çıkarılan veri modellerinde bulunan bilgilerin içeriği kişisel verileri içerirse, kişisel verilerin açığa çıkması gibi sorunlar meydana gelebilir.

Sonuç olarak teknoloji her ne kadar hayatımıza kolaylıklar getirse de bize getirdiği çok önemli dezavantajları ve çözülmeyi bekleyen problemleri de olabiliyor. İşte burada, veri madenciliği yapılacağı sırada kişilerin mahrem bilgilerinin saklanması çözülmesi gereken bir problemdir. Mahremiyeti Koruyan Veri Madenciliği, ifşa edilmesi istenmeyen veya izin verilmemiş hassas bilgileri korumak isteyen veri madenciliği alanıdır [6].

Mahremiyeti Koruyan Veri Madenciliği bir çok alanda uygulanabilir. Bunlardan bazıları şunlardır [6]:

- Anket ve Veri Toplama: Şirketlerin tavsiye ürünler sunabilmek için müşterilerinin kişisel tercihlerinin verilerini toplaması veya anketler yaparak iş planları oluşturması, politik partilerin stratejilerini ayarlamaları için kamuoyu yoklamaları yapmaları. Eğer ankete katılanların hassas verilerinin korunduğu kanıtlanabilirse, bu tarz anket verilerinin kapsamı önemli şekilde artabilir.

- Acil Durumları İzleme: Toplumun güvenliği veya ulusal güvenlik için potansiyel olağandışı durumları erkenden saptamak çok önemlidir.

- Ürün İzlenebilirliği: Yakın gelecekte tüm ürünler ve ambalaj üniteleri Radyo Frekans Tanımlama etiketi ile üretimden kullanıcıya ulaşıma kadar tüm adımlarda (paketleme, kargolanma vs.) otomatik olarak kaydedilecek. Bu büyük çapta bir veri birikimine yol açacak ve mahremiyet koruyucuları çok önemli olacak.

- Tıbbi Araştırma: Kişisel sağlık verileri en hassas olan mahrem bilgilerden biridir ve birçok devlet tarafından güvenlik standartları yasa haline getirilmiştir.

- Sosyal Ağlar: Sosyal ağlarda birçok bilgi kişiseldir, telif haklıdır ve Web'den uzak tutulmalıdır. Limitli ifşa etme seçeneklerine izin vermek için Mahremiyeti Koruyan Veri Madenciliği'nden yararlanılabilir.

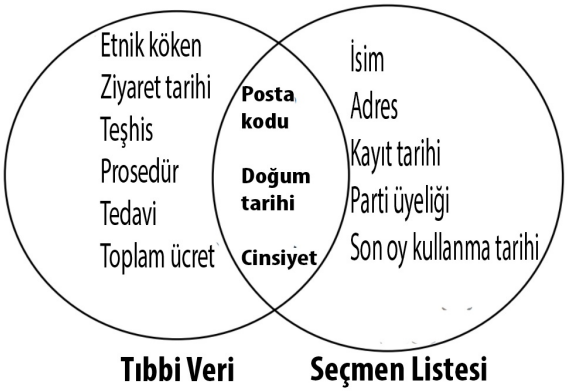
4.2 Yapılan Çalışmalar

4.2.1 *K*-Anonimlik

Genel bir uygulama olarak, verileri yayınlamada isim, adres ve telefon numarası gibi açıkça kimlik belirleyici alanlar çıkarılarak anonimliğin devamının sağlanması varsayılmıştır [7]. Ancak bağlama veya eşleştirme yöntemleriyle bu varsayımın gerçeği yansıtmadığı görülmüştür [8]. Örnek vermek gerekirse Sweeney tarafından yapılan çalışmada [8], 1990 nüfus sayım özet verileri kullanılarak bir deney yürütülmüştür. Ancak görülen sonuçlarda birkaç kombinasyonla benzersiz veya neredeyse benzersiz bir şekilde kimlik tespitinin yapıldığı görülmüştür [8]. 5-basamaklı posta kodu, cinsiyet ve doğum günü kombinasyonlarıyla Amerika Birleşik Devletleri popülasyonunun %87'sinin, yani 248 milyonda 216 milyon kişinin, büyük olasılıkla kimliğinin tespitinin yapılabildiği belirlenmiştir [8]. Sonuç olarak açık bir şekilde söylemek gerekirse açık kimlik belirtilen alanların çıkarılarak verilerin yayınlanmasının anonimliği sağlamadığı belirgindir.

Bağlama veya eşleştirme ile kimlik belirleme ve kişinin hassas verilerine ulaşılmasına örnek olarak Latanya Sweeney'in [7]'de yaptığı çalışmayı örnek verebiliriz: Massachusetts eyaletinde, Grup Sigorta Komisyonu eyalet çalışanlarının sağlık sigortalarını almakla sorumludur. Kimlik belirleyici alanlar çıkartılarak anonim hale geldiğine inandıklarından dolayı, komisyon topladığı yaklaşık 135 bin eyalet çalışanının ve ailelerinin kişisel verilerinin bulunduğu veri setinin bir kopyasını araştırmacılara kopyalayıp vermiş ve bir kopyasını da endüstriye satmıştır. Halbuki daha önce de belirtildiği üzere açık kimlik tanımlayıcıları çıkarmak anonimliği sağlamıyor. Yayınlanan bu verileri eşleştirebilmek için Latanya Sweeney Cambridge Massachusetts seçmen listesini 20 dolara satın alarak eşleştirme yapmaya çalışılmıştır. Ve sonuç olarak sadece posta kodu, doğum günü ve cinsiyet değerleri üzerinden bağlama yaparak kimlik belirleme yapıyor ve dolayısıyla kişilerin hassas verileri olan sağlık verilerine erişiyor. Hatta eyalet valisinin sağlık verilerine ulaşırken ilk önce

valiyle doğum günü aynı olan 6 kişi olduğunu buluyor. Bu 6 kişiden 3'ünün erkek 3'ünün kadın, doğum günü ve cinsiyet filtresinden sonra kalan 3 kişi içinde eyalet valisiyle aynı posta koduna sahip sadece bir eşleşme kalıyor ve dolayısıyla eyalet valisinin sağlık verilerine erişmiş oluyor. Şekil 4.1'de bağlamanın nasıl yapıldığı görselleştirilmiştir.



Şekil 4.1 : Bağlama ile yeniden kimlik tanımlama [7].

Latany Sweeney tarafından mahremiyeti korumaya yönelik olarak K-Anonimlik standardı önerilmiştir [7]. Bu yöntemde veri setindeki her bir bireyin kayıdı için en az k-1 adet farklı ayırt edilemeyen birey kayıtları bulunmalıdır.

Elimizde Çizelge 4.1'deki [9] gibi bir veri seti olduğunu düşünelim.

Çizelge 4.1 : Yatan Hasta Mikro Verisi [9].

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

Anonim hale getirilmeye çalışılan veri seti, Çizelge 4.2 [9] :

Çizelge 4.2 : *4*-Anonim Yatan Hasta Mikro Verisi [9].

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	130**	< 30	*	Heart Disease
2	130**	< 30	*	Heart Disease
3	130**	< 30	*	Viral Infection
4	130**	< 30	*	Viral Infection
5	1485*	≥ 40	*	Cancer
6	1485*	≥ 40	*	Heart Disease
7	1485*	≥ 40	*	Viral Infection
8	1485*	≥ 40	*	Viral Infection
9	130**	3*	*	Cancer
10	130**	3*	*	Cancer
11	130**	3*	*	Cancer
12	130**	3*	*	Cancer

Çizelge 4.2'deki veri Zip Code, Age ve Nationality kolonlarından dolayı 4-Anonimlik bir veri setidir ve bu şekilde anonimleştirilmiş bir veri setine bakan bir saldırgan, örnek olarak Çizelge 4.1'deki 8 numaralı kaydın bilgilerini bilen bir saldırgan olarak veri tabanına sorgular attığı zaman, 8 numaralı kayıdın sahip olabileceği Condition kolonunda hassas olan verinin Cancer, Heart Disease veya Viral Infection olacağını görür. Bu şekilde Saldırgan kişinin 8 numaralı kayıdın Condition kolonunda hassas olan verinin öğrenmesi zorlaştırılır.

Önerilen standard olan *K*-Anonimleştirme hassas veriyi ne kadar saklamaya çalışsa da problemleri mevcuttur ve güvenliği garanti etmemektedir [9]. Yukarıdaki Çizelge 4.2'den devam etmek gerekirse eğer ki saldırgan, Çizelge 4.1'deki 11 numaralı kaydın bilgilerini biliyorsa saldırıyı başarıyla gerçekleştirebilecek ve 11 numaralı kaydın Condition kolonunun değerini %100 öğrenebilecekti. Çünkü veri setinden dönen tüm satırlarda aynı hassas değer olan Cancer dönecektir. Dolayısıyla *k*-anonimleştirilmiş veri setleri yüzde yüz korumayı garanti etmiyor. Sonuç olarak bir veri kümesinde *k*-anonimlik sağlanmış olabilir ancak ilgili kayıtlar aynı hassas veriye sahip olurlarsa hassas veriler tespit edilebilir [9].

4.2.2 *L*-Çeşitlilik

K-Anonimleştirilmiş bir veri seti hassas verilerin korunmasını garanti etmemektedir [9]. *K*-anonimlik metodunun üstüne uygulanması gereken *L*-Çeşitlilik metodu geliştirilmiştir [9]. Bu yönteme göre bir veri setinde *K*-Anonimlik sağlandıktan sonra her bir grup için olan hassas veri değerlerinin çeşitlilik sağlanması gerekmektedir [9].

Çizelge 4.3 : *3*-Çeşitli Yatan Hasta Mikro Verisi [9].

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	1305*	≤ 30	*	Heart Disease
4	1305*	≤ 30	*	Viral Infection
9	1305*	≤ 30	*	Cancer
10	1305*	≤ 30	*	Cancer
5	1485*	> 40	*	Cancer
6	1485*	> 40	*	Heart Disease
7	1485*	> 40	*	Viral Infection
8	1485*	> 40	*	Viral Infection
2	1306*	≤ 40	*	Heart Disease
3	1306*	≤ 40	*	Viral Infection
11	1306*	≤ 40	*	Cancer
12	1306*	≤ 40	*	Cancer

L-Çeşitlilik prensibini açıklamak gerekirse [9], yarı belirleyicilerine göre ayrılmış her bir grup için en az *L* değeri kadar farklı sayıda iyi temsil edilmiş hassas verinin bulunmasını sağlayarak *L*-Çeşitlilik sağlanmaktır. Çizelge 4.3'de *3*-Çeşitlilik sağlanmıştır. Açıklamak gerekirse Zip Code değeri 1485*, Age değeri > 40 olan grup için hassas bilgiler olan Condition kolonu Cancer, Heart Disease, Viral Infection olmak üzere 3 farklı değer döner. Diğer gruplara bakıldığında da Condition kolonu 3 farklı değere sahiptir. Dolayısıyla bu *K*-Anonimlik sağlanmış veri setindeki tüm gruplar için değişik *L* kadar, 3, farklı hassas veri gözlemlendiği için Çizelge 4.3'de *L*-Diversity sağlanmıştır ve *3*-Çeşitlilik sağlanan bir veri setidir.

4.2.3 *T*-Yakınlık

Çizelge 4.4'de gösterilen ve *K*-Anonimlik ve *L*-Çeşitlilik sağlanmış tablonun 1, 2 ve 3 numaralı kayıtlardan oluşan gruba bakıldığında, bu gruba düşen herhangi bir kaydın hastalığının mide ile alakalı olduğu anlaşılır ve buna Benzerlik atağı denir [10].

Çizelge 4.4 : *3*-Çeşitlilik sağlanmış bir tablo [10].

	Zip Code	Age	Salary	Disease
1	476**	2*	3K	gastric ulser
2	476**	2*	4K	gastritis
3	476**	2*	5K	stomach cancer
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
7	476**	3*	7K	bronchitis
8	476**	3*	9K	pneumonia
9	476**	3*	10K	stomach cancer

Dolayısıyla K-Anonimlik ve L-Çeşitlilik mahremiyet korumada yeterli olmayabilir ve L-Çeşitlilik [10] yönteminin de bazı eksikleri vardır [10]. Bu açığın sebebi, L-Çeşitlilik her bir grup içindeki hassas verilerin çeşitliliğini sağlasa da, bu verilerin semantik olarak birbirleriyle yakınlıklarını hesaba katmamasıdır [10].

Kısaca belirtmek gerekirse, aynı seviyede çeşitlilik sağlanmış dağılımların çok farklı seviyede güvenlik sağlamalarının sebepleri şunlardır [10]:

- öznitelik değerleri arasında anlamsal ilişkiler mevcuttur,
- farklı değerlerin farklı hassasiyet değerleri mevcuttur,
- güvenlik ayrıca, bütün dağılım ile ilişkiden etkilenir.

Çizelge 4.5 : Salary Özniteliğine Göre 0.167-Yakınlık, Disease Özniteliğine Göre 0.278-Yakınlık Sağlanmış Tablo [10].

	Zip Code	Age	Salary	Disease
1	4767*	≤ 40	3K	gastric ulser
3	4767*	≤ 40	5K	stomach cancer
8	4767*	≤ 40	9K	pneumonia
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
2	4760*	≤ 40	4K	gastritis
7	4760*	≤ 40	7K	bronchitis
9	4760*	≤ 40	10K	stomach cancer

Bir sınıfın T-Yakınlık'a sahip olması için, sınıf içindeki hassas bir özniteliğin dağılımıyla, o özniteliğin tüm tablodaki dağılımı arasındaki farkın t'den fazla uzak

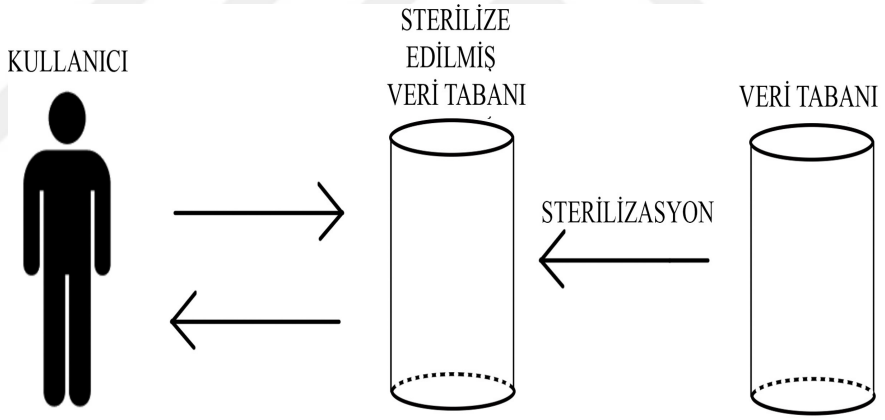
olmaması gerekmektedir [10]. Bir tablonun T-Yakınlık'a sahip olması için, tablo içindeki tüm sınıfların T-Yakınlık'a sahip olması gerekmektedir [10]. Çizelge 4.5'de [10] T-Yakınlık sağlanmış bir tablo gösterilmektedir.

5. DİFERANSİYEL MAHREMİYET

5.1 Diferansiyel Mahremiyet Nedir?

Etkileşimli ve etkileşimli olmayan olarak iki adet gizlilik mekanizması modeli mevcuttur [1]:

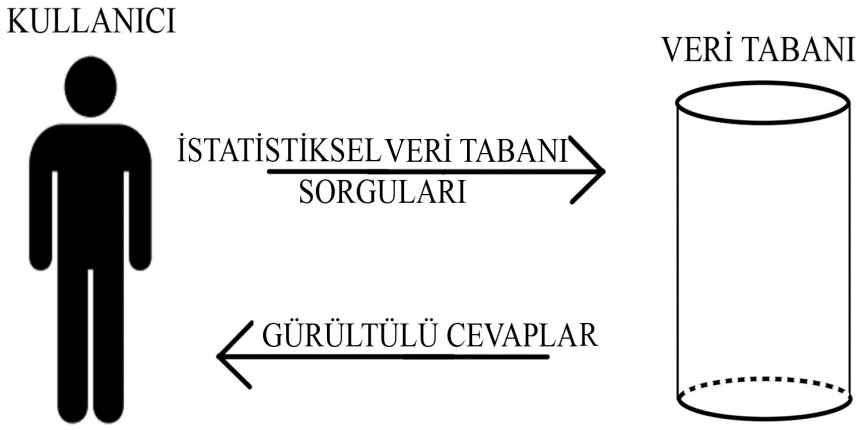
- Etkileşimli olmayan gizlilik mekanizması modeli: Bu yöntemde toplanan verinin sterilize edilmiş versiyonu güvenilir veri toplayıcısı tarafından yayımlanır. Geleneksel olarak veri setini sterilize etmek için veri değişimi ve seyrek örneklemenin yanı sıra isimler, doğum tarihleri, sosyal güvenlik numaraları gibi kimlik belirleyici alanları çıkarma teknikleri kullanılır. Örneğin K-Anonimlik [7] sağlanmış bir veri setinin veri analizcilerine paylaşılması (bkz : Çizelge 4.2).



Şekil 5.1 : Etkileşimli Olmayan Güvenlik Mekanizması Modeli.

- Etkileşimli güvenlik mekanizması modeli: Bu yöntemde güvenilir veri toplayıcısının sunduğu bir arayüz üzerinden kullanıcıların veritabanına sorgular atılması sağlanır ve muhtemelen gürültü eklenmiş cevaplar dönülür.

Diferansiyel Mahremiyet konsepti 2006 yılında Dwork tarafından ortaya atılmıştır [1] ve bu makalesinde istatistiksel veritabanı güvenliğinin, krpitoloji alanındaki semantik güvenliğin aksine, kişisel verilerin korunmasını garanti etmesinin mümkün olmadığını kanıtlamıştır. Semantik güvenli bir kriptolojik sistemde, görülmeden öğrenilemeyen şifreli mesaj incelenerek açık mesaj hakkında herhangi bir bilgi elde edinilemez.



Şekil 5.2 : Etkileşimli Güvenlik Mekanizması Modeli.

Ancak istatistiksel veritabanları için benzeri tanımların mahrem bilgileri koruma yöntemleri için mümkün olmadığı ispat edilmiştir. Bunun sebebi de saldırıyı yapacak kişinin yardımcı bilgiye sahip olmasıdır [1]. Örneğin [1] bir hassas verinin bir kişinin net boyunun olduğu ve bunun ortaya çıkmasının güvenlik ihlaline sebep olabileceği bir durumu düşünelim. Veritabanından farklı milletlerdeki kadınların boy ortalamasının çekildiğini farz edelim. Veri setine erişimi olan bir saldırgan yardımcı bir bilgi olarak, kişisel verisini öğrenmek istediği kişinin boyunun veri tabanından çekilen bir milletin boy ortalamasından bildiği bir miktar kadar kısa olduğunu bilirse, kişisel veri saldırgan tarafından bulunabilir. Diferansiyel Mahremiyet buna yönelik olarak ortaya çıkan bir güvenlik mekanizmasıdır.

Diferansiyel Mahremiyet yalnızca istatistiksel veritabanı sorgularına izin verir ve analiz sonucunun tek bir kayda bağlı olmamasını sağlamayı amaçlamaktadır [11]. Bunu sağlamak için tek bir kayıtta farklılık gösteren veritabanlarından elde edilecek analiz sonuçlarının güçlü olasılıkta aynı sonucu üretmesini sağlamaya çalışmaktadır.

Tanım 1 *(KOMŞU VERİTABANLARI). Aynı şemaya ve satır sayısına ait D ve D' veritabanları, sadece bir kayıtta birbirlerinden farklıysa bu veritabanlarına Komşu Veritabanları denir [11].*

Tanım 2 *(ε-DİFERANSİYEL MAHREMİYET). D ve D' nın tüm komşu veritabanları ve algoritma $S \subseteq Range(A)$'nın tüm olası sonuçları için olasılıkların A'nın rastgelelik derecesinin üzerinde olduğu yerlerde, rastgeleleştirilmiş A algoritması ε-Diferansiyel olarak gizlidir [11].*

$$Pr[A(D) \in S] \leq e^{\varepsilon} \times Pr[A(D') \in S] \tag{5.1}$$

Tanım 3 *($S_{L_1}(Q)$: Q'nun L_1 HASSASİYETİ). Varsayım olarak q(D)'nın q sorgusunun D veritabanındaki çıktısı olduğunu farz edelim. Q sorgu kümesinde, Q sorgu kümesinin hassasiyeti $S_{L_1}(Q)$ ile ifade edilerek, D ve D' veritabanlarının komşu olduğu yerlerde şu şekilde bulunur [11]:*

$$S_{L_1}(Q) = \max_{D,D'}\left(\sum_{q\in Q} |\, q(D) - q(D')\,|\right) \tag{5.2}$$

Laplace mekanizmasıyla gerçek cevaba gürültü eklenerek ε-Diferansiyel Mahremiyet sağlanabilir [11]. Gürültü ölçeği mahremiyet bütçesi olan ε ve sorgu kümesinin hassasiyetiyle hesaplanır.

Tanım 4 *(LAPLACE MEKANİZMASI). Lap(σ) ifadesinin, ortalaması 0 ve ölçek parametresinin σ olan Laplace dağılımından örneklenmiş rastgele bir değişken olduğunu farz edelim. A algoritması q sorguları için, q:D $\rightarrow$ R, $\lambda \geqslant S_{L_1(Q)} / \varepsilon$ olduğu durumlarda şu şekilde cevap verir $A(q,D) = q(D) + Lap(\lambda)$ [11].*

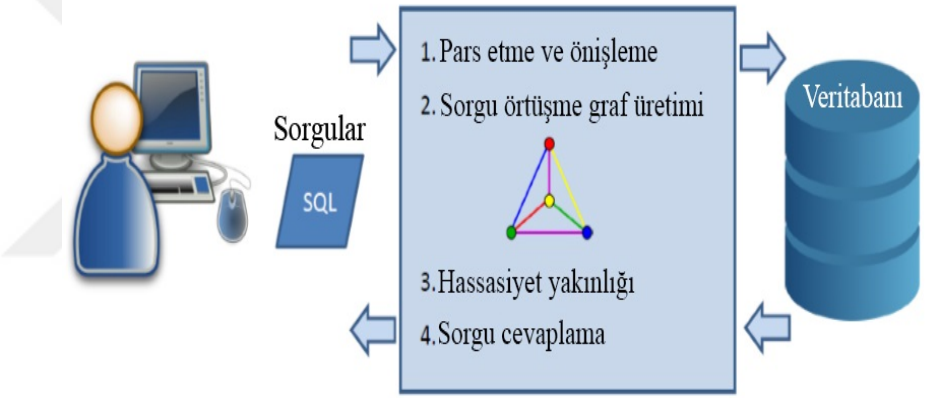
Tanım 4'de bulunan λ sembolü gürültü büyüklüğünü temsil etmektedir [11].

Formülden de görüleceği üzere eklenecek gürültü miktarı hassasiyetle doğru orantılı, mahremiyet bütçesiyle ters orantılıdır.

6. GRAF BAZLI SORGU SETİ YÖNTEMİ İLE DİFERANSİYEL MAHREMİYET

6.1 Yönteme Genel Bir Bakış

Diferansiyel Mahremiyet'te en önemli sorunlardan biri sorgu kümesinin hassasiyetini bulmaktır [11]. Sorgu kümesinin hassasiyetini bulmanın NP-Hard bir problem olduğu kanıtlanmıştır [12]. Bu tez çalışmasında [11, 13]'deki çalışmalar baz alınarak, sadece sunulan yönteme uygun istatistiki sorgular için alan bazlı olarak sorgu kümesinin yaklaşık olarak üst sınırdan hassasiyetinin bulunmasını sağlayarak nasıl Diferansiyel Mahremiyete erişilmeye çalışıldığı anlatıldı. Şekil 6.1'de yöntem gösterilmiştir [11].



Şekil 6.1 : Çözümün İş Akışı [11].

Yöntemi adım adım detayına girmeden açıklamak gerekirse [11];

1. İlk izlenmesi gereken yol, sorgular ileri bölümlerde anlatacağım standartlara göre filtrelenip, yönteme göre cevaplanabilinenler hassasiyet hesaplamasına dahil edilir, yönteme uygun olmayanlar ise elenir.

2. Sorguların alanları bulunur ve kesişimlerine göre grafa dökülür.

3. Yaklaşık hassasiyeti bulmak için graftaki maksimum klikte bulunan düğümlerin sayısı üzerinden hassasiyete erişilir.

4. Son olarak veritabanından dönen gerçek cevaplara Laplace dağılımıyla gürültü eklenir. Dağılımın ölçeği üst sınır hassasiyet değeri ve mahremiyet değeri olan ε ile belirlenir.

6.2 COUNT'a Göre Sorgu Modeli

Referans makaledeki [11] yöntemle Diferansiyel Mahremiyet SQL dilinde uygulanabilir ve Toplama Fonksiyonlarından bazılarına (COUNT, SUM, MIN ve MAX) uygulanabilir. Ancak bu tezimde sadece COUNT Toplama Fonksiyonu üzerinden ilerleyeceğim. Makalede de belirtildiği üzere [11] COUNT sorgusu ile dörtgen alanlarıyla oluşturulan sorgu kümeleri bir çok veri analizi yöntemleri için elverişlidir.

Yönteme uygun sorgu modellerinden bahsetmek gerekirse [11]:

- İstatistiksel olmayan sorgular elenir.
- Yöntem aynı anda sadece tek bir tabloya uygulanabilir. Yani hassasiyet hesaplanırken atılan sorgular JOIN olmayan sadece tek bir tabloya atılmalıdır. Dolayısıyla tek d-boyutlu tablo T hassasiyet hesabına katılmalıdır.
- Tablodaki her bir kolon veya özelliği Q_1,Q_2,....Q_d ile ifade edersek domainleri sınırlı olmalıdır. Sonsuz olursa yönteme uygun olmaz.
- Öznitelikler nümerik, kategorik veya sıralı olabilir. String tipindeki özelliklerse filtrelenerek elenir.
- WHERE sözcüğünden sonra gelen karşılaştırma betiminde iki özelliğin olduğu bir fonksiyon olursa elenir.
- WHERE sözcüğünden sonra gelen karşılaştırma betiminde aynı özellik birden fazla kullanılırsa elenir.
- Karşılaştırma betiminde operatör olarak sadece AND, BETWEEN, $=$, $>$, $<$, $\geq$, $\leq$ kullanılır.
- GROUP BY sözcüğüne izin verilmez.

Yukarıdaki standartlara göre oluşturulan sorgular sayesinde hiper dikdörtgenler [11] elde edilerek alanların kesişimi üzerinden sorguları grafa dökme sağlanır.

Standarda uymayan sorgu örnekleri;

- Sorgu 1 : *SELECT age FROM customers* `customer_id` *< 10*
  Bu sorgu sintaks hatası vereceğinden elenir.
- Sorgu 2 : *SELECT COUNT(*) FROM customers, transactions*
  Bir sorguda sadece bir tabloya sorgu atılabilir.
- Sorgu 3 : *SELECT COUNT(*) FROM (SELECT customer_id FROM customers WHERE* `customer_id` *BETWEEN 5 and 14) AS T*
  İç içe sorgular yönteme uygun değildir.
- Sorgu 4 : *SELECT COUNT(*) FROM customers WHERE* `customer_name` *LIKE %Emir´*
  LIKE sözcüğü içeren sorgular yönteme uygun değildir.
- Sorgu 5 : *SELECT COUNT(*) FROM customers WHERE* `customer_id` *< 10 OR* `customer_id` *> 23*
  OR karşılaştırma betimi yöntemin standartlarına uygun olmadığı için elenir.
- Sorgu 6 : *SELECT age FROM customers*
  Bu sorgu istatistiksel bir sorgu olmadığı için elenir.
- Sorgu 7 : *SELECT COUNT(*) FROM customers WHERE* `customer_id` *!= 24*
  Sorgular eşit değildir operatörleriyle veritabanına gönderilemez.
- Sorgu 8 : *SELECT COUNT(*) FROM customers WHERE* `customer_id` *<> 24*
  Sorgular eşit değildir operatörleriyle veritabanına gönderilemez.
- Sorgu 9 : *SELECT COUNT(*) FROM customers WHERE* `customer_id` *NOT IN (2,3,6)*
  Sorgular NOT IN operatörleriyle veritabanına gönderilemez.
- Sorgu 10 : *SELECT COUNT(*) FROM customers WHERE* `customer_id` *IN (2,3,6)*
  IN operatörü sorgularda kullanılamaz.
- Sorgu 11 : *SELECT COUNT(*) FROM customers WHERE* `customer_id` *= 24 AND* `customer_id` *< 11*

WHERE sözcüğünden sonra gelen karşılaştırma betiminde bir kolon bir kereden fazla kullanılamaz.

- Sorgu 12 : *SELECT COUNT(*) FROM customers WHERE* `customer_id` *> GREATEST(2,77,3,677)*
  WHERE sözcüğünden sonra gelen karşılaştırma betiminde herhangi bir fonksiyon kullanılamaz.

- Sorgu 13 : *SELECT COUNT(*) FROM customers GROUP BY* `income`
  GROUP BY sorgularda kullanılamaz.

Standarda uyan örnek bir sorgu kümesi:

- Sorgu 1 : *SELECT COUNT(*) FROM customers WHERE* `user_id` *BETWEEN 1923 AND 2023*
- Sorgu 2 : *SELECT COUNT(*) FROM customers WHERE age* $\geq$ *23*
- Sorgu 3 : *SELECT COUNT(*) FROM customers WHERE age < 55 AND* `user_id` *> 1923*
- Sorgu 4 : *SELECT COUNT(*) FROM customers*
- Sorgu 5 : *SELECT COUNT(*) FROM customers WHERE income < 30K*
- Sorgu 6 : *SELECT COUNT(*) FROM customers WHERE age > 40*
- Sorgu 7 : *SELECT COUNT(*) FROM customers WHERE age* $\leq$ *55 AND income > 10K*
- Sorgu 8 : *SELECT COUNT(*) FROM customers WHERE income = 10K*
- Sorgu 9 : *SELECT COUNT(*) FROM customers WHERE* `user_id` *> 10*
- Sorgu 10 : *SELECT COUNT(*) FROM customers WHERE age* $\leq$ *55 AND income > 10K AND* `user_id` *= 210*

6.3 Sorguların Alanlarını Belirleme

Bu aşamaya gelen sorgular bir önceki bölümde açıkladığım yönteme uygun standarttaki istatiksel sorgulardır. *D*-boyutlu bir tablo *T*'nin oluşturacağı *d*-boyutlu hiper dikdörtgenler oluşur [11].

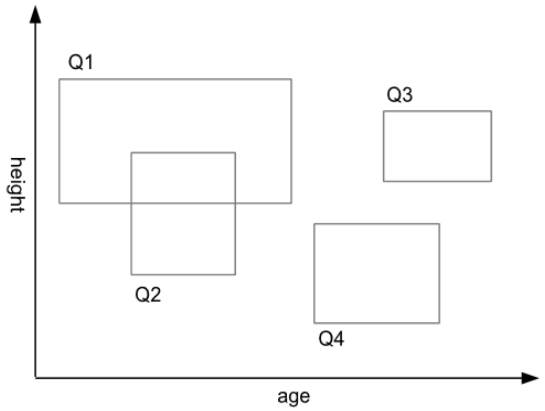
Yönteme uygun standartlardan oluşan bir sorgu kümesinin oluşturduğu hiper dikdörtgenlere bir örnek vermek için aşağıdaki sorgu kümesinden açıklama yapmaya çalışacağım. *T*'nin *2*-boyutlu bir tablo olduğu varsayılarak *Age* ve *Height* gibi iki özniteliği mevcuttur. Bu tablodan cevap almak için atılan aşağıdaki sorgu kümesinin, *Q*'dan elenen ve yönteme uyan sorgu kümesine, *Q'*, dönüştürüldükten sonra atıldığını varsayalım [11]:

- *Q*1 : *SELECT COUNT(*) FROM T WHERE Age BETWEEN 5 AND 30 AND Height BETWEEN 160 AND 190*
- *Q*2 : *SELECT COUNT(*) FROM T WHERE Age BETWEEN 15 AND 25 AND Height BETWEEN 130 AND 170*
- *Q*3 : *SELECT COUNT(*) FROM T WHERE Age BETWEEN 40 AND 50 AND Height BETWEEN 165 AND 185*
- *Q*4 : *SELECT SUM(Age) FROM T WHERE Age BETWEEN 35 AND 45 AND Height BETWEEN 110 AND 155*

Yönteme göre alanlarına göre oluşturulan hiper dikdörtgenler Şekil 6.2'deki gibidir [11].

Tanım 5 *Bir sorgular kümesinin alan-kesişimi. Eleman sayısı 1'den fazla olan,* $|Q_S| > 1$*, bir sorgu kümesi* Q_S *için, alan-kesişimi* $alan_{Q_S}$ *ile ifade edilir ve* Q_S *içindeki tüm sorguların alanlarının kapsamına giren bir alandır [11].*

$$alan_{Q_S} = \bigcap_{q \in Q_S} alan_q \tag{6.1}$$



Şekil 6.2 : *Q'* İçindeki Sorguların Alanları [11].

Örneğin Şekil 6.2'de, Q_S = {$Q1$, $Q4$} için kesişim aralığı boş kümedir. Ancak Q_S = {$Q1$, $Q2$} için kesişim aralığı mevcuttur.

Algoritma 1'in uygulanmasıyla (INTERSECTS) iki sorgunun birbiriyle kesişip kesişmediğini bulabiliriz [11].

Algoritma 1 Comparing regions of queries p and q [11]

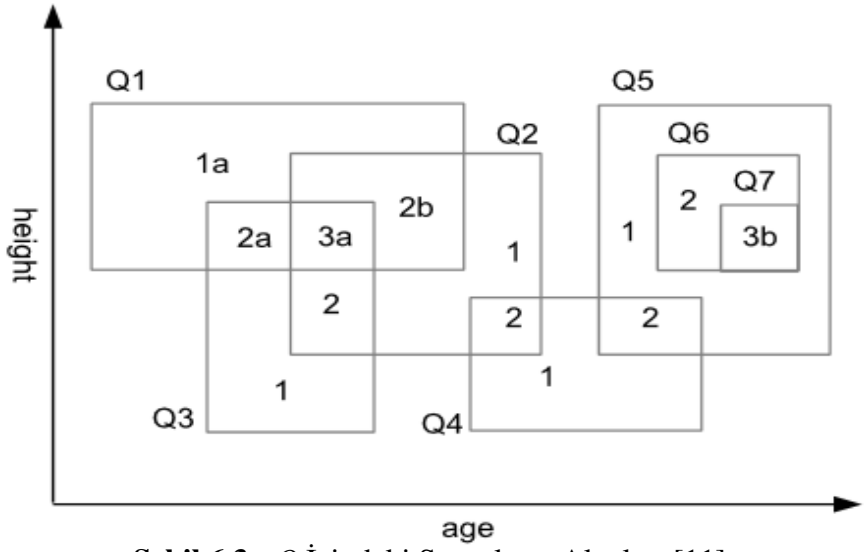
```
1: function INTERSECTS(Query p, Query q)
2:     for Each att. A_i listed in both p.where
         and q.where do
3:         range_p^{A_i} ← p.where[A_i]
4:         range_q^{A_i} ← q.where[A_i]
5:         if range_p^{A_i} ∩ range_q^{A_i} = ∅ then
6:             return false
7:     return true
```

Algoritma 1'in hesaplama karmaşıklığı *O(d)* kadardır [11]. Verilen örnek için T tablosunun boyutu olan 2 , yani $O(2)$ kadardır.

6.4 Grafa Dökme

Alan bazlı sorguların kesişim hiper dikdörtgenleri çizildikten sonra Algoritma 2 (GEN-GRAPH) uygulanarak sorgu kümesi grafa dökülür ve daha sonrasında maksimum klik bulunması için bir sonraki adıma geçilir [11]. Yine referans makalesinden örnek vermek gerekirse bir sorgu kümesini ve bu sorgu kümesine göre Şekil 6.3 hiper dikdörtgenler oluştuğunu varsayalım [11]:

Aynı sorgu kümesinden sorgu alanlarının kesişimlerine göre graf oluşturmak için Algoritma 2 (GEN-GRAPH) uygulanır [11]. Bu algoritmaya göre, grafta sorgu sayısı



Şekil 6.3 : Q İçindeki Sorguların Alanları [11].

kadar düğüm oluşturulur ve her bir düğüm bir sorguyu temsil eder. Bu sorgular içinde birbirleriyle kesişen sorgular varsa bunlarda kenarlarla birbirlerine bağlanırlar.

Hesaplama karmaşıklığı tablo boyutunun d, sorgu sayısının S olduğu zaman $O(d \text{ x } |S|^2)$ olan Algoritma 2'yi uygulayarak, Şekil 6.3'teki sorgu alanlarından oluşan hiper dikdörtgenlere göre graf oluşturduğumuzda Şekil 6.4 elde edilir.

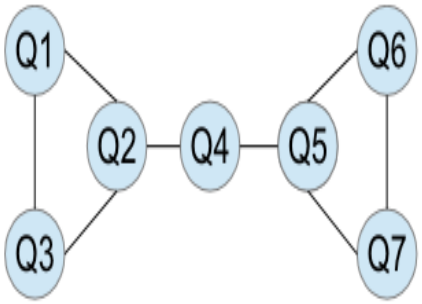
Algoritma 2 Mapping Q to $G(V,E)$ [11]

```
1:  function GEN-GRAPH(Query set Q)
2:      V ← ∅
3:      for Each query q ∈ Q do
4:          V ← V ∪ {q}
5:      E ← ∅
6:      for Each query p ∈ Q do
7:          for Each query q ∈ Q, p ≠ q do
8:              if INTERSECTS(p,q) then
9:                  E ← E ∪ {(p,q)}
10:     return G(V,E)
```

Şekil 6.4'te de görüleceği üzere tüm sorgular bir düğümle temsil edilmiş, aralarında kesişen sorgular bir kenarla birbirlerine bağlanmış ve graf oluşturulmuştur. Bundan sonraki adım yaklaşık hassasiyeti bulmak için graftaki maksimum kliki bulmaktır.

6.5 Yaklaşık Hassasiyet Bulma

Referans alınan Ali İnan, Mehmet Emre Gürsoy, Emir Esmerdağ ve Yücel Saygın'ın çalışmalarında [11, 13], teorik olarak hassasiyetin üst sınırının, sorguları



Şekil 6.4 : Q'dan Haritası Çizilmiş Graf [11].

grafa döktükten sonra maksimum klik problemiyle çözülebileceği kanıtlanmıştır. Maksimum klik, bir grafta birbirleriyle bağlantısı olan en büyük sayıdaki düğümler topluluğudur ve bir grafta birden fazla maksimum klik mevcut olabilir [11]. Şekil 6.4'ten göstermek gerekirse, Q1, Q2, Q3 ve Q5, Q6, Q7 bu graftaki maksimum kliklerdir.

Graf Bazlı Sorgu Seti Yöntemi ile sorgu kümesinin hassasiyetini bulurken, oluşturulan graftaki maksimum klikte bulunan eleman sayısı iki ile çarpılır ve hassasiyet elde edilir [11]. Ancak maksimum klikte bulunan eleman sayısının iki ile çarpımı grafta bulunan tüm düğümlerin sayısından fazla ise graftaki düğüm sayısı hassasiyet olarak ele alınır. Maksimum klikte bulunan düğüm sayısı MKS, Grafta bulunan düğüm sayısı N olarak farz edilirse [11]:

$$S_{L_1}(Q) < min\{N, (2 \times MKS)\} \tag{6.2}$$

Sorgu kümesinin hassasiyetinin bulunması için algoritma 3 (APPROX-SENS) uygulanmalıdır [11]:

Algoritma 3 Approximating $S_{L_1}(Q)$ [11]

1: **function** APPROX-SENS(Query set Q)
2: $G \leftarrow$GEN-GRAPH(Q)
3: $MCS \leftarrow MCS(G)$
4: **return** $\min(|Q|, 2 \times MCS)$

6.6 Yöntemin Uygulanması İçin Örnekler

Bu bölümde, Graf Bazlı Sorgu Seti Yöntemi ile hassasiyet bularak diferansiyel mahremiyetin sağlanması için oluşturulan yöntem için sürecin baştan sona örnekleri verilecektir.

6.6.1 Örnek 1

Adımları sıralamak gerekirse:

1. Kullanıcının erişimi olduğu bir ara yüz aracılığıyla epsilon değerini 0.01 olarak girdiğini ve aşağıdaki sorgu setini, $S1$, girdiğini farz edelim;
    - $Q1$: *SELECT COUNT(*) FROM customers WHERE age = 19*
    - $Q2$: *SELECT COUNT(*) FROM cities WHERE city_id > 11*
    - $Q3$: *SELECT COUNT(*) FROM cities WHERE country_id* $\leq$ *23*
    - $Q4$: *SELECT COUNT(*) FROM customers WHERE age < 19*
    - $Q5$: *SELECT COUNT(*) FROM customers WHERE user_id > 1000 AND age < 18*
2. Graf Bazlı Sorgu Seti Yöntemi ile hassasiyetin bulunması için sorgu setinde atılan tüm sorguların hepsinin aynı tabloya sorgu atması beklenmektedir. Ancak bu örnekte görüldüğü gibi $Q2$ ve $Q3$ sorguları *cities* tablosuna sorgu atarken, $Q1$, $Q4$ ve $Q5$ sorguları *customers* tablosuna sorgu atmaktadır. Dolayısıyla graf bile oluşturulmadan bu sorguların hiç birine cevap verilmeyecektir. Olası bir kullanıcı arayüzünde gerekli bilgi mesajı yayınlanabilir.

6.6.2 Örnek 2

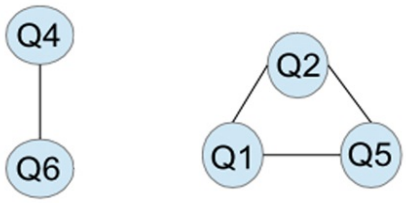
Adımları sıralamak gerekirse:

1. Kullanıcının erişimi olduğu bir ara yüz aracılığıyla epsilon değerini 0.3 olarak girdiğini ve aşağıdaki sorgu setini, $S2$, girdiğini farz edelim;
    - $Q1$: *SELECT COUNT(*) FROM customers WHERE age BETWEEN 33 AND 66*
    - $Q2$: *SELECT COUNT(*) FROM customers WHERE age > 20 and user_id > 1923*
    - $Q3$: *SELECT user_name FROM customers WHERE user_id = 1991*

- $Q4$: *SELECT COUNT(*) FROM customers WHERE user_id > 1000 AND age < 18*
- $Q5$: *SELECT COUNT(*) FROM customers WHERE age* $\geq$ *23*
- $Q6$: *SELECT COUNT(*) FROM customers WHERE age < 19*

2. Graf Bazlı Sorgu Seti Yöntemi ile hassasiyetin bulunmasına uygun olmayan sorgular ilk olarak elenmelidir. Önceki bölümlerde açıkladığım istatistiksel olmayan veya yönteme uymayan sorgular elenir. $Q3$ burada yönteme uymayan sorgudur ve elenmesi gerekir. Dolayısıyla hassasiyet hesabına $Q3$ dahil edilmez.

3. Daha sonra oluşturulan sorgu alanları belirlenip kesişimleri bulunan sorgu seti $S2$ için Şekil 6.5 gözüken graf oluşturulur.



Şekil 6.5 : $S2$ Sorgu Setinden Oluşturulan Graf.

4. Artık graf oluşturulduğuna göre şimdiki adım sorgu kümesinin hassasiyetini bulmaktır. Bunun içinde Algoritma 3'teki yöntem (APPROX-SENS [11]) uygulanır. Yani oluşturulan graftaki maksimum klikte bulunan eleman sayısı bulunur ve iki ile çarpılır. Eğer elde edilen değer graftaki tüm kliklerin sayısından fazla ise, graftaki klik sayısı, değilse çarpım sonucu elde edilen değer üst sınırlı hassasiyet değeri olarak ele alınır. Yani bu örnek için maksimum klikte bulunan eleman sayısı 3 olduğundan 3 x 2 değeri de graftaki tüm düğümlerin sayısı olan 5'ten büyük olduğundan üst sınırlı hassasiyet değeri 5 olarak ele alınıp hesaplamalara sokulur.

5. Son adım olarak veritabanından dönen değerlere, Laplace mekanizmasıyla, hassasiyet/epsilon ile ölçeklenecek şekilde gürültüler yuvarlanarak eklenir. En son aşamada da kullanıcı sadece gürültülü cevaplara erişebilmiş olur ve dolayısıyla ε-Diferansiyel Mahremiyet sağlanarak analiz yapılmış olur.

6.6.3 Örnek 3

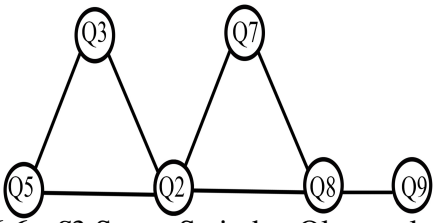
Adımları sıralamak gerekirse:

1. Kullanıcının erişimi olduğu bir arayüz aracılığıyla epsilon değerini 0.1 olarak girdiğini ve aşağıdaki sorgu setini, $S3$, girdiğini farz edelim;

   - $Q1$: *SELECT COUNT(*) FROM citizens,countries*
   - $Q2$: *SELECT COUNT(*) FROM citizens WHERE citizen_id* $\leq$ *1000*
   - $Q3$: *SELECT COUNT(*) FROM citizens WHERE district_id = 5 AND citizen_id < 500*
   - $Q4$: *SELECT COUNT(*) FROM citizens WHERE citizen_id != 15*
   - $Q5$: *SELECT COUNT(*) FROM citizens WHERE district_id > 3 AND citizen_id < 500*
   - $Q6$: *SELECT COUNT(*) FROM citizens WHERE district_id != 24*
   - $Q7$: *SELECT COUNT(*) FROM citizens WHERE citizen_id BETWEEN 750 AND 1000*
   - $Q8$: *SELECT COUNT(*) FROM citizens WHERE citizen_id > 750 AND district_id < 3*
   - $Q9$: *SELECT COUNT(*) FROM citizens WHERE citizen_id > 1000 AND district_id = 1*

2. Graf Bazlı Sorgu Seti Yöntemi ile hassasiyetin bulunmasına uygun olmayan sorgular ilk olarak elenmesi gerektiğinden burada birden fazla tabloyu aynı sorguda kullandığı için sorgu $Q1$, eşit değildir operatörünü kullandıkları için $Q4$ ve $Q6$ sorgularının elenmesi gerekir. Yani graf oluşturma aşamasında $Q1$, $Q4$ ve $Q6$ işlenmeyerek hassasiyet hesabına dahil edilmeyecek, dolayısıyla bu sorgulara da cevap verilmeyecektir.

3. Daha sonra oluşturulan sistem tarafından sorgu alanları belirlenerek aşağıdaki resimde ortaya çıkan graf oluşturulur.



Şekil 6.6 : $S3$ Sorgu Setinden Oluşturulan Graf.

4. Graf oluşturulduktan sonraki adım sorgu kümesinin hassasiyetini bulmaktır [11]. Bunun için de Algoritma 3'teki yöntem (APPROX-SENS) uygulanır. Bu örnek için maksimum klikte bulunan eleman sayısı 3 olduğundan 3 x 2 değeri de graftaki tüm düğümlerin sayısı olan 7'ten küçük olduğundan üst sınırlı hassasiyet değeri 6 olarak ele alınıp hesaplamalara sokulur.

5. Son adım olarak veritabanından dönen değerlere, Laplace mekanizması, hassasiyet/epsilon ile ölçeklenecek şekilde gürültüler yuvarlanarak eklenir. En son aşamada da kullanıcı sadece gürültülü cevaplara erişebilmiş olur ve dolayısıyla ε-Diferansiyel Mahremiyet sağlanarak analiz yapılmış olur.

6.6.4 Örnek 4

Adımları sıralamak gerekirse:

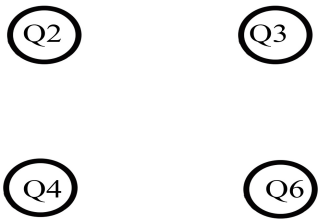
1. Kullanıcının erişimi olduğu bir ara yüz aracılığıyla epsilon değerini 0.6 olarak girdiğini ve aşağıdaki sorgu setini, $S4$, girdiğini farz edelim;

   - $Q1$: *SELECT COUNT(*) FROM (SELECT student_id FROM students WHERE* `student_id` *BETWEEN 5 and 14) AS S*
   - $Q2$: *SELECT COUNT(*) FROM students WHERE student_id* $\leq$ *10*
   - $Q3$: *SELECT COUNT(*) FROM students WHERE student_id = 12 AND age < 15*
   - $Q4$: *SELECT COUNT(*) FROM students WHERE student_id > 15 AND age = 13*
   - $Q5$: *SELECT COUNT(*) FROM students WHERE* `student_name` *LIKE %Jonhson´*
   - $Q6$: *SELECT COUNT(*) FROM students WHERE age* $\geq$ *16 AND student_id = 15*

- *Q*7 : *SELECT COUNT(*) FROM students WHERE* `student_id` *IN (2,3,6)*

2. Yine bu örnekte de diğer örneklerde olduğu gibi Graf Bazlı Sorgu Seti Yöntemi ile hassasiyetin bulunmasına uygun olmayan sorgular ilk olarak elenmesi gerektiğinden iç içe sorguları kullandığı için sorgu *Q*1, LIKE sözcüğünü kullandığı için *Q*5, IN operatörünü kullandığı için *Q*7 sorgularının elenmesi gerekir. Yani graf oluşturma aşamasında *Q*1, *Q*5 ve *Q*7 işlenmeyerek hassasiyet hesabına dahil edilmeyecek, dolayısıyla bu sorgulara cevap verilmeyecektir.

3. Daha sonra oluşturulan sistem tarafından sorgu alanları belirlenerek aşağıdaki resimde ortaya çıkan graf oluşturulur.



Şekil 6.7 : *S*4 Sorgu Setinden Oluşturulan Graf.

4. Algoritma 3 uygulanarak bulunan hassasiyet şu şekildedir. Grafta görüldüğü üzere hiç bir eleman birbirlerine bağlanan hiçbir düğüm gözükmemektedir. Yani sorguda hassasiyet 1 x 2 sonucunda elde edilen rakam olan 2'dir. Bir diğer dikkat edilmesi gereken ise 2'nin grafta bulunan tüm düğümlerin sayısından az olduğudur.

5. Yine bir önceki örneklerde olduğu gibi son adım olarak veritabanından dönen gerçek değerlere, Laplace Mekanizması uygulanarak hassasiyet/epsilon ile ölçeklenecek şekilde gürültü eklenerek ε-Diferansiyel Mahremiyet sağlanarak veri analizi yapılmış olur.

7. VERİ ANALİZİ YÖNTEMLERİNDE GRAF BAZLI SORGU SETİ YÖNTEMİ İLE HASSASİYET BULARAK DİFERANSİYEL MAHREMİYETİN SAĞLANMASI

7.1 Öznitelik Seçimi

Öznitelik Seçimi, model oluşturulmadan evvel alt kümeleri seçme sürecidir ve Graf Bazlı Sorgu Seti Yöntemiyle Diferansiyel Mahremiyet sağlanarak, öznitelikler arası entropinin hesaplanmasıyla üst-*k* Öznitelik Seçimi sağlanabilir [13]. Bu entropi değerlerine göre sıralama yapmak için, özniteliğine göre en az entropi değeri bulunan öznitelik en üst sırada bulunan öznitelik olarak sıralanır [13]. Bunun için veritabanına entropi hesaplamaları yaparken, arayüzden atılan sorgular Graf Bazlı Sorgu Seti Yöntemiyle Diferansiyel Mahremiyet sağlamaya uygun olarak atılmalı, ve dönen gerçek cevaplara muhtemel gürültü eklenerek özniteliklerin entropi hesaplarının yapılması gerekir [13]. Bir önceki bölümde anlatılan yönteme uygun sorgu setleri oluşturularak bir çok analiz görevleri mahremiyet korunmaya çalışarak uygulanabilir.

7.1.1 Bilgi kazanımıyla entropi hesaplama

Bilgi Kazanımı, Öznitelik seçimi ölçümüdür ve aşağıda açıklanan yöntemle *D* içindeki bir demetin sınıflandırılabilmesi için beklenen bilgi miktarı ölçülebilir [14]:

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i) \tag{7.1}$$

Info(D) ile gösterilen *D*'nin entropisidir.

$Info_A(D)$ ise, *v* kadar farklı değere sahip *A* özniteliğine göre paylaştırılmış *D* içindeki bir demetin sınıflandırılabilmesi için beklenen bilgi miktarıdır ve şu şekilde bulunur [14]:

$$Info_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j) \tag{7.2}$$

A özniteliğine göre paylaştırılmış *D* için bilgi kazancı yukarıdaki iki değer bulunduktan sonra aşağıdaki formül ile bulunabilir ve bulunan değer *A* dan paylaştırma yaptığımız zaman ne kadar kazancımız olduğunu gösterir [14]:

$$Gain(A) = Info(D) - Info_A(D) \tag{7.3}$$

Bu sürece bir örnek vermek için bir müşteri veri setini (Çizelge 7.1) ele alalım [14]:

Çizelge 7.1 : *AllElectronics* Müşteri Veritabanından Elde Edilmiş Sınıf-Etiketli Eğitim Demetleri [14].

RID	age	income	student	credit_rating	Class:buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Tabloda ele aldığımız veri setinde *buys_computer* özniteliği sınıflandırıcıdır. Buna göre ilk başta bulunması gereken *D* içindeki bir demetin sınıflandırılabilmesi için beklenen bilgi miktarı yani *Info(D)* dir. Bu tabloya göre elde edilen sonuç ise şu şekildedir;

$$Info(D) = -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14}) = 0.940\,bits. \tag{7.4}$$

Bir öznitelik olarak *age* özniteliğine göre paylaştırılmış *D* içindeki bir demetin sınıflandırılabilmesi için beklenen bilgi miktarını bulmak için $Info_{age}(D)$ bulunmalıdır.

$$Info_{age}(D) = \frac{5}{14} \times (-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}) + \frac{4}{14} \times (-\frac{4}{4}\log_2\frac{4}{4}) + \frac{5}{14} \times (-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}) = 0.694\ bits. \quad (7.5)$$

En son olarak *age* özniteliğine göre paylaştırma yapılırsa, elde edilecek bilgi kazanımını buluruz:

$$Gain(age) = Info(D) - Info_{age}(D) = 0.940 - 0.694 = 0.246\ bits. \quad (7.6)$$

Buna benzer olarak, veri setindeki diğer özniteliklere göre paylaştırma yapılırsa elde edilecek bilgi kazançları da ölçülür. *Gain(income)* = 0.029 bits, *Gain(student)* = 0.151 bits ve *Gain(credit rating)* = 0.048 bits. Bu veri setinde en çok bilgi kazancı sağlayacak olan öznitelik yani *age* seçilmelidir.

Graf Bazlı Sorgu Seti Yöntemiyle Diferansiyel Mahremiyet sağlayarak bu işlemleri yapmak için [13] özniteliklerin bilgi kazançlarını ölçme sırasında veritabanına atılan sorgulardan (COUNT istatistiksel ve yönteme uygun sorgular) gelen cevaba gürültü eklenmelidir. Böylece Diferansiyel Mahremiyet sağlanarak Öznitelik Seçimi işlemi gerçekleştirilebilir ve en çok bilgi kazancı olan veya en az entropi değeri olan sıralamada en üstte olacak şekilde öznitelik sıralaması yapılabilir.

7.2 Korelasyon Analizi

Gereğinden fazla öznitelikleri çıkarma veri entegrasyonunda bir diğer sorundur ve bir öznitelik başka bir öznitelik kümesiyle elde edilebilirse gereksiz olabilir [15]. Bu sorunu çözmek için Korelasyon Analizi kullanılabilir. Kategorik veriler için χ^2 (ki-kare) testi uygulanabilir [15]. Bu analiz işlemini yaparken de Graf Bazlı Sorgu Seti Yöntemiyle Diferansiyel Mahremiyet sağlamak için, yani analiz için veri tabanına atılan ve yönteme uygun olan COUNT sorgularından dönen cevaplara muhtemel gürültü eklemek gerekmektedir [13].

7.2.1 Ki-Kare testi ile korelasyon analizi

Ki-kare testi ile kategorik olan iki farklı özniteliğin birbirleriyle korelasyonu bulunabilir [15]. Bunu hesaplamak için olumsallık tablosundan gidilir ve keşisen değerlerde beklenen ile elde edilen sayılara göre iki öznitelik arasındaki korelasyon belirlenir. Böylece iki özniteliğin birbirlerine bağlı olup olmadığı anlaşılır.

Ki-kare test ile Korelasyon Analizine bir örnek vermek için aşağıdaki ihtimal tablosunu, Çizelge 7.2, ele alalım [15]:

Çizelge 7.2 : 2 x 2 Olumsallık Tablo Verisi [15].

	male	**female**	**Total**
fiction	250(90)	200(360)	450
non_fiction	50(210)	1000(840)	1050
Total	300	1200	1500

Çizelge 7.2'ye göre iki öznitelik arasındaki korelasyon χ^2 ile şu şekilde bulunur [15]:

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840} \tag{7.7}$$

$$= 284.44 + 121.90 + 71.11 + 30.48 = 507.93.$$

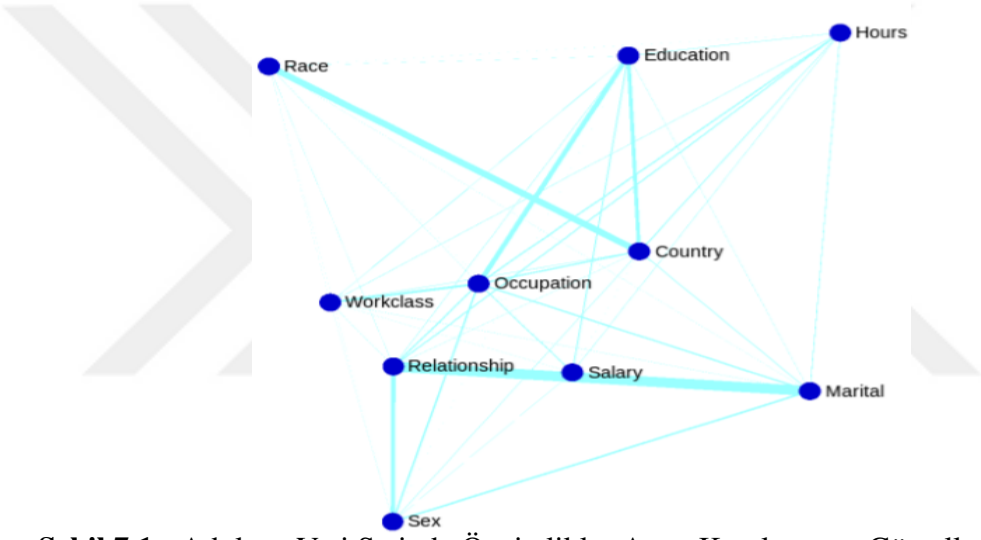
Çıkan sonuca göre iki öznitelik arasındaki ki-kare değerinin, bu tablo için serbestlik derecesi (2-1) x (2-1) = 1 iken, Ki-Kare tablosunda 1 serbestlik derecesinde 0.001 önem düzeyindeki 10.828 değerinden yüksek olduğunu görürüz ve sonuç olarak bu özniteliklerden ikisinden biri gereğinden fazla öznitelik olarak çıkarılabilir [15].

Kesişimler için atılan sorgular COUNT sorgularıda ve karşılaştırma betimleri AND ile devam etmektedir [13]. Öznitelikler kategorik olduğu içinde Graf Bazlı Sorgu Seti Yöntemiyle Diferansiyel Mahremiyet sağlanabilir. Çizelge 7.2'a göre örnek vermek gerekirse;

- *SELECT COUNT(*) FROM Table WHERE gender = 'male' AND preferred_reading='fiction'* sorgusundan dönen değer tabloya göre 250 olsada hassasiyet ve epsilon değerine bağlı olarak sorguya gürültü eklenebilir.

- *SELECT COUNT(*) FROM Table WHERE gender = 'female'* sorgusundan dönen gerçek değer 1200'e gürültü eklenebilir.
- *SELECT COUNT(*) FROM Table* sorgusu kamu bilgisi olarak göz önünde bulundurulmuştur, dolayısıyla bu tarz Graf Bazlı Sorgu Seti Yöntemine uygun COUNT fonksiyouyla oluşturulmuş istatiksel sorgular için gürültü eklenmemektedir [13].

Sonuç olarak belirtmek gerekirse Ki-kare test için veritabanına atılacak sorgu seti, Graf Bazlı Sorgu Seti Yöntemiyle Diferansiyel Mahremiyet sağlamak için sunulan yönteme uyduğu takdirde Diferansiyel Mahremiyet sağlanarak analizi yapılabilir. Örnek olarak Şekil 7.1'de öznitelikler arası korelasyon graf üzerinden görselleştirilmiştir [13].



Şekil 7.1 : Adultset Veri Setinde Öznitelikler Arası Korelasyonu Görselleştirme [13].

7.3 Sınıflandırma

Naïve Bayes sınıflandırıcılar istatistiksel sınıflandırıcılardır ve bir veri demetinin hangi sınıfa ait olma olasılığını hesaplayarak sınıflandırma yapar [16] .

Bayes teorisini detaylı açıklamak gerekirse [16], X'in bir veri demeti olduğunu farz edelim, yani Bayes teorisinde bir kanıt. Bir hipotez olarak X'in bağlı olduğu bir sınıfın C olduğu hipotezi H olarak ele alalım. Sınıflandırma problemleri için $P(H|X)$ yani X veri demetinin gözlemlendiği yerde H gerçekleşme olasılığı ele alınır.

X veri demetinin gözlemlendiği yerde H gerçekleşme olasılığı için $P(H|X)$ sonsal olasılıktır [16] ve Örnek vermek gerekirse [16], müşterilerin yaş, gelir ve bilgisayar alıp almadığı verilerinin olduğu bir veri tabanı düşünelim. 35 yaşında, 40 bin

dolar geliri olan bir müşterinin X olduğunu farz edelim. Hipotezimiz, H, olarak da müşterinin bilgisayar alması durumunu farz edelim. 35 yaşında, 40 bin dolar geliri olan X müşterinin bilgisayar alma olasılığı $P(H|X)$ olarak tanımlanır ve bu sonsal olasılıktır.

$P(H)$ ise önsel olasılıktır [16] ve veri tabanında herhangi bir müşterinin bilgisayar alma olasılığını yansıtır.

Benzer olarak, $P(X|H)$ ise sonsal olasılıktır [16] ve bilgisayar alan müşterinin 35 yaşında ve 40 bin doları olma olasılığını verir. $P(X)$ ise bir önsel olasılıktır ve veri tabanındaki müşterilerin 35 yaşında ve 40 bin dolar geliri olma olasılığını verir.

Bayes Teoremi sonsal olasılığı bulmak için uygundur ve *Bayes Teoremi* şu şekildedir [16]:

$$P(H \mid X) = \frac{P(X \mid H)\, P(H)}{P(X)} \tag{7.8}$$

7.3.1 Naïve bayes sınıflandırıcı

Aşağıdaki maddelerden yola çıkarak açıklamak gerekirse [16]:

- X'in n boyutlu bir öznitelik vektörü olduğunu farz edelim, $X = (x_1, x_2, \ldots. x_n)$,
- C'nin m adet farklı sınıfı olduğunu farz edelim, $C_1, C_2, \ldots\ldots C_m$
- C_i sınıfının X'in gözlemlendiği yerde gözlemlenme olasılığı *Bayes Teoreminden* bu formül ile elde edilir:

$$P(C_i \mid X) = \frac{P(X \mid C_i)\, P(C_i)}{P(X)} \tag{7.9}$$

  Ancak bütün C sınıfları için $P(X)$ aynı olacağı için $P(C_i|X)$'in maksimumlaştırılmış değeri alınarak, yani $P(X)$ hesaplamalardan çıkarılarak işlemler yapılır.

- $P(X|C_i)$ yani C_i koşulunun olduğu yerde X'in gözlemlenme olasılığını bulmak için X değerlerinin birbirlerinden koşullu bağımsız olduğunu farz ederiz ve aşağıdaki formülden buluruz:

$$P(X \mid C_i) = \prod_{k=1}^{n} P(X_k \mid C_i) \tag{7.10}$$

Naïve Bayes sınıflandırıcılar ile Sınıflandırma işlemine örnek vermek için [16] Çizelge 7.1'deki veri setini ele alalım. Örnek tabloda *age*, *income*, *student* ve *credit_rating* olmak üzere dört adet öznitelik ve *yes* ve *no* olmak üzere iki farklı değer alan *buys_computer* sınıflandırıcı özniteliği bulunmaktadır. Bu eğitim veri setine göre *age* öznitelik değerinin *youth*, *income* öznitelik değerinin *medium*, *student* öznitelik değerinin *yes* ve son olarak da *credit_rating* öznitelik değerinin *fair* olduğu bir kullanıcının bilgisayar alıp almama olasılıklarını hesaplayıp sınıflandırma yapmak istersek *Naive Bayes sınıflandırıcı* yöntemine göre aşağıdaki şekilde bulabiliriz [16];

$$
\begin{aligned}
P(buys\_computer = yes) &= 9/14 = 0.643 \\
P(buys\_computer = no) &= 5/14 = 0.357 \\
P(age = youth \mid buys\_computer = yes) &= 2/9 = 0.222 \\
P(age = youth \mid buys\_computer = no) &= 3/5 = 0.600 \\
P(income = medium \mid buys\_computer = yes) &= 4/9 = 0.444 \\
P(income = medium \mid buys\_computer = no) &= 2/5 = 0.400 \\
P(student = yes \mid buys\_computer = yes) &= 6/9 = 0.667 \\
P(student = yes \mid buys\_computer = no) &= 1/5 = 0.200 \\
P(credit\_rating = fair \mid buys\_computer = yes) &= 6/9 = 0.667 \\
P(credit\_rating = fair \mid buys\_computer = no) &= 2/5 = 0.400
\end{aligned}
\tag{7.11}
$$

Bu olasılıklar kullanılarak $P(X \mid buys\_computer = yes)$ değerini bulmak için aşağıdaki formül uygulanır:

$$
\begin{aligned}
P(X \mid buys\_computer = yes) &= P(age = youth | buys\_computer = yes) \\
&\times P(income = medium \mid buys\_computer = yes) \\
&\times P(student = yes \mid buys\_computer = yes) \\
&\times P(credit\_rating = fair \mid buys\_computer = yes) \\
&= 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044
\end{aligned}
\tag{7.12}
$$

Benzer olarak $P(X \mid buys\_computer = no)$ için şu sonucu elde ederiz:

$$P(X \mid buys\_computer = no) = 0.600 \times 0.400 \times 0.200 \times 0.400 = 0.019 \qquad (7.13)$$

En son olarak kullanıcının olası bilgisayar alma ve almamasına karar vermek için yani sınıflandırmak için aşağıdaki formülü uygularız:

$$\begin{aligned} P(X \mid buys\_computer = yes)P(buys\_computer = yes) &= 0.044 \times 0.643 = 0.028 \\ P(X \mid buys\_computer = no)P(buys\_computer = no) &= 0.019 \times 0.357 = 0.007 \end{aligned} \qquad (7.14)$$

Sonuç olarak çıkan maksimumlaştırılmış değerlere göre kullanıcının bilgisayar alma olasılığı daha yüksektir. *Naïve Bayes sınıflandırması* yöntemi bu şekildedir. Bu analizde de Graf Bazlı Sorgu Seti Yöntemiyle Diferansiyel Mahremiyet sağlanabilir [13]. Bunun için de veri tabanına atılan ve yönteme uygun olan COUNT sorgularından dönen cevaplara muhtemel gürültü eklemek gerekmektedir. *Naïve Bayes sınıflandırması* için veritabanına atılacak sorgular Graf Bazlı Sorgu Seti Yönteminin belirlediği sorgu standartlarına uyarak yapılabilir.

8. SONUÇ

Bilgi çağında yaşadığımız söylensede bizler aslında veri çağının içinde bulunmaktayız [3]. Veri çağında yaşayan bizler için veriden bilgiyi çıkarmak bir gereksinim olmuştur ve bu iş için gereken süreçlerden biri de Veri madenciliğidir [4]. Büyük çapta veriler için gerekli bilgi çıkarılması için verilerin veri analizcilerine sunulması, kişilerin hassas verilerinin ortaya çıkması gibi bir durumu meydana getirmiştir. Bu dezavantajın engellenmesi için Mahremiyeti Koruyan Veri Madenciligi alanı gelişmiştir ve ifşa edilmesi istenmeyen veya izin verilmemiş hassas bilgileri korumak isteyen veri madenciligi alanıdır [6]. Bu alanda K-Anonimlik [7], L-Çeşitlilik [9], T-Yakınlık [10] gibi yöntemler geliştirilmiştir. Ayrıca Dwork Diferansiyel Mahremiyet'i anlattığı makalesiyle [1] anlattığı yöntemde etkileşimli olan bir güvenlik mekanizması yöntemi sunulmuştur. Dwork bu makalesinde, saldırıyı yapacak kişinin yardımcı bilgiye sahip olmasına dayanarak, istatistiksel veritabanı güvenliğinde kişisel verilerin korunmasının garanti edilmesinin mümkün olmadığını kanıtlamıştır.

Bu tezin yazılmasında [11, 13]'deki çalışmalardan yola çıkarak, istatistiksel sorgular üzerinden Diferansiyel Mahremiyet'in SQL dilinde uygulanabileceği yöntem anlatılmaya çalışılmıştır. Ayrıca bu yöntem, veri analizi yöntemlerine de aynı şekilde uygulanabilir [13].

8.1 Çalışmanın Uygulama Alanı

Bu çalışmanın uygulanabileceği alan veri paylaşımının olduğu her alan olabilir. Özellikle devlet kurumlarının araştırma ve geliştirme projelerinde kullanılmak üzere veri paylaşımı gerçekleşebilir, burada vatandaşların hassas verilerinin korunması önem taşımaktadır. Başka bir kullanım alanı olarak şirketlerin veri madenciliği yaparak bilgi keşfetmesi gereken işlerinde analizcinin veritabanındaki hassas verileri öğrenmesine bir engel olması için bu yöntem kullanılabilir. Bu yöntemi uygulamak için gerekli

bir arayüz sağlandıktan sonra analizcilerin yönteme uygun sorguları atması sağlanarak muhtemel gürültülü cevaplar yansıtılabilir.

8.2 Öneriler

Yöntem bazı sorgu tiplerine izin vermektedir [11]. Dolayısıyla bu yöntem üzerinden yapılan analizlere, yeni sorgu modelleri eklenebilir. Yaygın olarak kullanılan SQL dilinde oluşturulabilecek tüm istatistiksel sorgular için uygunluk sağlamaya yönelik çalışmalar yapılabilir.

KAYNAKLAR

[1] **Dwork, C.** (2006). Differential Privacy, *Proceedings of the 33rd International Conference on Automata, Languages and Programming - Volume Part II*, ICALP'06, Springer-Verlag, Berlin, Heidelberg, s.1–12, `http://dx.doi.org/10.1007/11787006_1`.

[2] **Szalay, A. ve Gray, J.** (2002). The world-wide telescope, an archetype for online science.

[3] **Han, J.**, **Pei, J. ve Kamber, M.** (2011). *Data mining: concepts and techniques*, Elsevier, 3. sürüm, sy. 1, 2.

[4] **Han, J.**, **Pei, J. ve Kamber, M.** (2011). *Data mining: concepts and techniques*, Elsevier, 3. sürüm, sy. 5, 6, 7, 8.

[5] **Weiss, G.M. ve Davison, B.D.**, (2010), Data Mining. Handbook of Technology Management, H. Bidgoli.

[6] **Evfimievski, A. ve Grandison, T.** (2009). Privacy preserving data mining, *IGI Global*, 1–8.

[7] **Sweeney, L.** (2002). k-anonymity: A model for protecting privacy, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *10*(05), 557–570.

[8] **Sweeney, L.** (2000). Simple demographics often identify people uniquely, *Health (San Francisco)*, *671*, 1–34.

[9] **Machanavajjhala, A.**, **Gehrke, J.**, **Kifer, D. ve Venkitasubramaniam, M.** (2006). l-diversity: Privacy beyond k-anonymity, *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*, IEEE, s.24–24.

[10] **Li, N.**, **Li, T. ve Venkatasubramanian, S.** (2007). t-closeness: Privacy beyond k-anonymity and l-diversity, *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, IEEE, s.106–115.

[11] **Inan, A.**, **Gursoy, M.E.**, **Esmerdag, E. ve Saygin, Y.** (2016). Graph-based modelling of query sets for differential privacy, *Proceedings of the 28th International Conference on Scientific and Statistical Database Management*, ACM, s. 3.

[12] **Xiao, X. ve Tao, Y.** (2008). Output perturbation with query relaxation, *Proceedings of the VLDB Endowment*, *1*(1), 857–869.

[13] **Esmerdag, E.**, **Gursoy, M.E.**, **Inan, A. ve Saygin, Y.** (2016). Explode: An Extensible Platform for Differentially Private Data Analysis, *Data Mining Workshops (ICDMW), 2016 IEEE 16th International Conference on*, IEEE, s.1300–1303.

[14] **Han, J.**, **Pei, J. ve Kamber, M.** (2011). *Data mining: concepts and techniques*, Elsevier, 3. sürüm, sy. 336, 337, 338, 339.

[15] **Han, J.**, **Pei, J. ve Kamber, M.** (2011). *Data mining: concepts and techniques*, Elsevier, 3. sürüm, sy. 94, 95, 96.

[16] **Han, J.**, **Pei, J. ve Kamber, M.** (2011). *Data mining: concepts and techniques*, Elsevier, 3. sürüm, sy. 350, 351, 352, 353, 354.

ÖZGEÇMİŞ

Ad Soyad: Emir Esmerdağ

Doğum Tarihi ve Yeri: 02.01.1991 İstanbul

E-Posta: esmerdag15@itu.edu.tr

ÖĞRENİM DURUMU:

- **Lisans:** 2014, Işık Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği

YAYINLAR:

- Emir Esmerdag, Mehmet Emre Gursoy, Ali Inan, and Yucel Saygin (2016). Explode: An Extensible Platform for Differentially Private Data Analysis, Data Mining Workshops (ICDMW), 2016 IEEE 16th International Conference on, IEEE, s.1300-1303.

- Ali Inan, Mehmet Emre Gursoy, Emir Esmerdag, and Yucel Saygin (2016). Graph-based modelling of query sets for differential privacy Proceedings of the 28th International Conference on Scientific and Statistical Database Management, ACM, s.3.