DeepL
Subscribe to DeepL Pro to translate larger documents.
Visit www.DeepL.com/pro for more information.

MARMARAUNIVERSITY GRADUATESTIFLERİNİNSTITUTE SİNDEKİPLANLAR

DATAPRIVASYONUNUNETKİLERİNEİLİŞ KİNKOMİSYONLARI MACHINELEARNINGALGORITHMSINIOT

TAJ ELDEEN SALEH

MASTERTHESIS

Elektronik ve Bilgisayar Mühendisliği Bölümü

DANIŞMAN

Doç. Dr. Ömer Korçak

ISTANBUL,2022

MARMARAUNIVERSITY
GRADUATESTIFLERİNİNSTITUTE
SİNEİLİŞKİNLİKLERİ

DATAPRIVASYONUNUNETKİLERİNEİLİŞ KİNKOMİSYONLARI MACHINELEARNINGALGORITHMSINIOT

TAJ ELDEEN SALEH

MASTERTHESIS

Elektronik ve Bilgisayar Mühendisliği Bölümü

DANIŞMAN

Doç. Dr. Ömer Korçak

ISTANBUL,2022

YAZARLIK BEYANI

Ben, Taj Eldeen Saleh, "IoT'de Veri Gizliliğini Koruma Yöntemlerinin Makine Öğrenmesi Algoritmaları Üzerindeki Etkilerinin Karşılaştırılması" başlıklı bu tezin ve içinde sunulan çalışmanın bana ait olduğunu beyan ederim. Bunu onaylıyorum:

- Bu çalışma tamamen veya esas olarak bu Üniversitede bir araştırma derecesi için adaylık sırasında yapılmıştır.

- Bu tezin herhangi bir bölümünün daha önce bu Üniversite veya başka bir kurumda bir derece veya başka bir yeterlilik için sunulmuş olması halinde, bu durum açıkça belirtilmiştir.

- Başkalarının yayınlanmış çalışmalarına başvurduğum durumlarda, bu her zaman açıkça belirtilmiştir.

- Başkalarının çalışmalarından alıntı yaptığım yerlerde, her zaman kaynak belirtilmiştir. Bu tür alıntılar haricinde, bu tez tamamen kendi çalışmamdır.

- Tüm ana yardım kaynaklarına teşekkür ettim.

- Tezin başkalarıyla birlikte yaptığım çalışmalara dayandığı durumlarda, başkaları tarafından yapılanları ve kendimin katkıda bulunduklarını tam olarak açıkladım.

Çalışmamı, bilgi arayışında her zaman ilham kaynağım olan bir baba, bir dede ve bir öğretmen olan büyükbabam Shukri Saleh'in anısına ithaf etmek istiyorum.

TEŞEKKÜR

Her şeyden önce, eğitim yolculuğumdaki sonsuz ve sürekli destekleri ve cesaretlendirmeleri için ebeveynlerime en derin minnettarlığımı ifade etmek isterim, ayrıca iki kız kardeşimin tüm desteği için minnettarım.

Çalışmamı keşfetmemi ve araştırmamı sağlayan ve teşvik eden ve bana çok takdir edilen içgörüler sağlayan, sürekli desteğini ve yardımını sunan tez danışmanım Prof.

Dr. Ali Çakmak'tan bahsetmek istiyorum; yüksek lisansımın büyük bir bölümünde biyoinformatik laboratuvarında birlikte çalıştım ve bu yolculukta bana gerçekten yardımcı olan araştırma ve deneyim kazanma şansına sahip oldum.

Son olarak, süreç boyunca yanımda olan, bana her zaman inanan ve beni neşelendiren arkadaşlarıma özellikle teşekkür etmek istiyorum, sizler olmadan bunu başaramazdım.

İÇİNDEKİLER

ÖZET

NESNELERİN İNTERNETİNDE VERİ GİZLİLİĞİNİ KORUMA YÖNTEMLERİNİN MAKİNE ÖĞRENME ALGORİTMALARINA ETKİLERİNİN KARŞILAŞTIRILMASI

Veri gizliliğini korumak, birçok kuruluş ve birey için çok önemli ve artan bir endişe kaynağıdır. Gizlilik konusunu ele almak için, veriye dayalı hizmetler araştırma ve geliştirme üzerinde doğrudan etkileri olan birçok düzenleme uygulanmaktadır. Verilerin anonimleştirilmesi, belirli gizlilik düzenlemelerine uymak için kişisel olarak tanımlanabilir bilgileri kaldırarak bu sorunla başa çıkmanın bir yoludur. Ancak, anonimleştirme süreci tek başına verilere bir miktar gürültü getirir.

Bu çalışmada, anonimleştirme algoritmalarının uygulanmasının makine öğrenmesi modellerinin performansı üzerindeki etkilerini anlamayı amaçlıyoruz. K-anonimliği ve l- çeşitliliği ve t-yakınlığı gibi farklı varyasyonlarını sağlamanın etkilerini bir dizi sınıflandırıcı ve gerçek hayat veri kümesi üzerinde karşılaştırıyoruz. Karşılaştırmalarımızda, makine öğrenimi için özelleşmiş bir bilgi kaybı metriği kullanıyoruz. Ayrıca, bilgi kaybını en aza indiren ve ***k-anonimlik*** özelliğini uygulayan optimal genelleme hiyerarşi ağaçlarını oluşturabilen ve seçebilen otomatik bir genelleme ve bastırma çerçevesi sunuyoruz. Sonuçlarımız, her k-anonimlik varyasyonunun farklı bir gizlilik düzeyi sunduğunu ve anonimleştirme sürecinde farklı kısıtlamalar getirdiğini göstermektedir. Genel olarak, anonimleştirme sürecinde ne kadar fazla kısıtlamamız olursa, verilerde o kadar fazla gürültü alırız.

Ayrıca, kullanıcıların ham verilerini toplamadan veya paylaşmadan ML modellerinin merkezi olmayan bir şekilde eğitilmesine izin veren federe öğrenme isimli yeni bir başka yaklaşımı da araştırdık. ***K-anonimleştirilmiş*** verileri kullanmaya adapte olabilen, silolar arası federe bir öğrenme çerçevesi tasarladık. Veri anonimleştirme entegrasyonunun daha iyi gizlilik sağlarken minimum bilgi kaybı sağlayabileceğini ve her iki yaklaşımı tek bir çerçevede kullanmanın her iki yaklaşımın avantajlarından yararlanmamızı sağladığını gösteriyoruz.

ÖZET

VERİ GİZLİLİĞİNİ KORUMA YÖNTEMLERİNİN IOT'DA MAKİNE ÖĞRENME ALGORİTMALARINA ETKİLERİNİN KARŞILAŞTIRILMASI

Veri gizliliğinin korunması, birçok kuruluş ve birey için çok önemli ve artan bir endişe kaynağıdır. Gizlilik konusunu ele almak için, veri odaklı hizmet araştırma ve geliştirme üzerinde doğrudan etkileri olan birçok düzenleme uygulanmaktadır. Veri anonimleştirme, belirli gizlilik düzenlemelerine uymak için kişisel tanımlanabilir bilgileri kaldırarak bu sorunla başa çıkmanın bir yoludur. Ancak, anonimleştirme işlemi kendi başına verilere bir miktar gürültü katmaktadır.

Bu çalışmada, anonimleştirme algoritmalarının uygulanmasının makine öğrenimi modellerinin performansı üzerindeki etkilerini anlamayı amaçlıyoruz. K-anonimlik ve farklı varyasyonlarının (l-çeşitlilik ve t-yakınlık olarak bilinir) uygulanmasının etkilerini bir dizi sınıflandırıcı ve gerçek hayat veri kümeleri üzerinde karşılaştırıyoruz. Karşılaştırmalarımızda, makine öğrenimi için özelleşmiş bir bilgi kaybı metriği kullanıyoruz. Ayrıca, bilgi kaybını en aza indiren ve k-anonimlik özelliğini uygulayan en uygun genelleştirme hiyerarşi ağaçlarını oluşturabilen ve seçebilen otomatik bir genelleştirme ve bastırma çerçevesi sunuyoruz. Sonuçlarımız, her bir ***k-anonimlik*** varyasyonunun farklı bir gizlilik seviyesi sunduğunu ve anonimleştirme sürecine farklı kısıtlamalar getirdiğini göstermektedir. Genel olarak, anonimleştirme sürecinde ne kadar çok kısıtlama olursa, verilerde o kadar çok gürültü elde ederiz.

Ayrıca, kullanıcıların ham verilerini toplamadan veya paylaşmadan makine öğrenimi modellerinin merkezi olmayan bir şekilde eğitilmesine olanak tanıyan başka bir yeni yaklaşım olan federe öğrenmeyi de araştırdık. K-anonimleştirilmiş verileri kullanmaya adapte olabilen bir çapraz-silo federe öğrenme çerçevesi tasarladık. Veri anonimleştirmenin entegrasyonunun daha iyi gizlilik sağlarken minimum bilgi kaybı sağlayabileceğini ve her iki yaklaşımın tek bir çerçevede kullanılmasının her iki yaklaşımın avantajlarından yararlanmamızı sağladığını gösteriyoruz.

SEMBOLLER

H : Koşullu Entropi

I : Entropi katsayısı

S : Belirli bir veri kümesi için sütun kümesi

C_j : Belirli bir sütundaki benzersiz değerler kümesi

T : Hedef sütundaki benzersiz değerler kümesi

ρ : Yüzdelik Aralık Değeri

f : Birleştirilmiş değerlerin sıklığı

R : Mondrian yaklaştırılmış kayıtlar kümesi

L_i : Bir kayıttaki satır kümesi

E_i : k kriterini karşılayan bir kaydın alt kümesi

A_i : k kriterini karşılamayan kayıtların alt kümesi

KISALTMALAR

ML : Makine Öğrenimi

FL : Federated Learning

DP : Veri Gizliliği

TFDS : TensorFlow veri oluşturucu

TP : Gerçek Pozitif

TN : Gerçek Negatif

FP : Yanlış Pozitif

FN : Yanlış Negatif

QI : Yarı Tanımlayıcılar

GDPR : Genel Veri Koruma Yönetmeliği

KVKK : Kişisel Verileri Koruma Kurumu

VGH : Değer Genelleştirme Hiyerarşisi

DGH : Etki Alanı Genelleştirme Hiyerarşisi

RAKAMLAR LISTESI

TABLOLARININ LISTESI

1. GİRİŞ

İçinde bulunduğumuz dijital çağda, akıllı telefonlar ve IoT cihazları gibi günlük hayatımıza entegre edilen yeni teknolojilerin kullanımıyla toplanan, depolanan ve paylaşılan veri miktarında benzeri görülmemiş bir artış yaşanmaktadır. Doğal olarak bu verilerin büyük bir kısmı kimlik, maaş, adres ve sağlık durumu gibi önemli miktarda kişisel bilgi içermektedir. Bu veriler şirketler tarafından araştırma, kişiselleştirilmiş reklamlar ve öneri sistemleri gibi hizmetler gibi çok çeşitli amaçlarla sürekli olarak toplanmakta ve işlenmektedir. Bu büyümeyle birlikte, verilerin güvenliğinin sağlanması ve gizli tutulmasıyla ilgili muazzam zorluklarla karşılaşıyoruz. Buradaki en büyük zorluk, verilere yalnızca yetkili tarafların erişimine izin verirken, verilerin kullanışlılığını korumaya çalışmaktır.

Bu zorluklar göz önünde bulundurulduğunda, birçok ülke bireysel gizlilik konusuyla ilgilenmeye başlamıştır. AB'de Genel Veri Koruma Yönetmeliği (GDPR) [1] ve Türkiye'de Kişisel Verileri Koruma Kurumu (KVKK) [2] gibi modern düzenlemeler ve standartlar ortaya çıkmıştır. Bu tür düzenlemeler, verilerin izinsiz toplanmasını, işlenmesini ve paylaşılmasını önlemek için dikkatlice gözden geçirilmekte ve uygulanmaktadır. Bu tür düzenlemeler ve yasalar, kişisel bilgilerin gizliliğini korumak için konulmuştur, ancak aynı zamanda hizmetlerini oluştururken ve sunarken buna bağlı olan birçok kuruluş için temel verilerin kullanılabilirliğini ve erişilebilirliğini de sınırlar. Bu nedenle, veri anonimleştirme [3], bir dereceye kadar veri kullanışlılığını korurken bu düzenlemelere uymak için yoğun bir şekilde kullanılmaktadır.

Veri anonimleştirme, bireylerden toplanan verilerin, tanımlayıcılarının kaldırılması veya şifrelenmesi yoluyla artık asıl sahiplerine kadar izlenememesini sağlayan bir süreçtir. Çok sayıda farklı ***anonimleştirme*** modeli bulunmakla birlikte, en yaygın olanı, benzer yarı tanımlayıcılara sahip veri kayıtlarının birlikte gruplandırılmasını sağlayan ***k-anonimliktir*** [4]; burada her grup, yarı tanımlayıcı değerlerinin tam olarak aynı dizisini içeren en az ***k-1*** örnek içerir. Bu özelliği sağlamak için genelleme, bastırma ve toplama gibi veri manipülasyon teknikleri kullanılır. Ancak, bu tür manipülasyonlar bir

Makine öğrenimi (ML) modellerinin performansını etkileyen önemli bilgi kaybı [5].

Anonimleştirme ile ilgili birçok çalışma olmasına rağmen, bunların çoğu anonimleştirme sürecinin optimizasyonunu araştırmakta ve çok azı bu tür yöntemlerin sınıflandırıcı performansı üzerindeki etkilerini araştırmaktadır. Biz bu çalışmada bir adım daha ileri giderek k-anonimliğin farklı varyasyonları olan l-çeşitlilik [6] ve t-yakınlık ile etkilerini araştırıyor ve karşılaştırıyoruz
[7] bir dizi sınıflandırıcı ve dört gerçek hayat veri kümesi üzerinde. Ayrıca bir tam sınıflandırıcı verilerin genelleştirilmesi ve bastırılması [8] için otomatik ve optimize edilmiş bir yöntemdir. Buna ek olarak, anonimleştirmenin veri tahmin performansı üzerindeki etkilerini anlamak için entropi tabanlı bir bilgi kaybı metriği kullanıyoruz. Bu, eğitimden önce makine öğrenimi performansını tahmin etmek için yeni ve kullanışlı bir yaklaşımdır, önceki yöntemler ise öncelikle anonimleştirme yoluyla yapılan genelleştirme seviyelerinin sayısını sayan metrikleri kullanmıştır.

Yakın zamanda Google araştırma grubu tarafından tanıtılan bir diğer yaklaşım ise, tüm veri işlemenin kullanıcı düzeyinde gerçekleştiği ve ham verilerin toplanmadığı veya paylaşılmadığı, bunun yerine sonuçlanan modellerin ağırlıklarının merkezi bir sunucu ile paylaşılacağı makine öğrenimi uygulamaları için birleştirilmiş öğrenmedir (FL) [35]. Hem veri anonimleştirme hem de FL, gizliliği potansiyel olarak tehlikeye atabilecek bir dizi zorluk ve sınırlamayla karşı karşıyadır. Bu çalışmada ayrıca ***k-anonimleştirmenin*** federe öğrenme çerçevelerine, daha spesifik olarak da çapraz-silo federe öğrenme çerçevesine entegrasyonunu araştırdık [36].
. Mahremiyetin korunmasına yönelik bir adım daha ileri gittiğine inandığımız önerilen çerçeve için ayrıntılı bir açıklama sunuyoruz. Çerçevemizi ***k-anonimleştirilmiş*** ve anonimleştirilmemiş veriler üzerinde test ederek aralarında uygun bir karşılaştırma yapıyoruz. Araştırmamız, anonimleştirme sürecinde genelleştirmenin uygun şekilde kullanılmasıyla verilerin kullanışlılığının korunabileceğini göstermektedir. Çerçeve sonuçlarımız, ***k-anonimlik*** entegrasyonunun, ***anonimleştirme*** nedeniyle modellerin performansı üzerinde sınırlı etkilerle gizlilik saldırılarına karşı daha iyi koruma sağladığını göstermektedir.

1.1 Önceki Çalışmalar

Verilerin nasıl anonimleştirileceği konusunu ele alan önemli miktarda çalışma bulunmaktadır [6-10]. Makine öğrenmesi tekniklerinin anonimleştirme sürecine entegrasyonu üzerine bazı çalışmalar mevcuttur [11-13]. Belirli veri kümeleri ve IoT alanları için anonimleştirme üzerine bazı çalışmalar yapılmıştır [14-16], ayrıca bazı makaleler genel amaçlı ve özel bilgi metriklerini anonimleştirme sürecine uyarlama üzerine çalışmıştır [6, 18]. Anonimleştirmenin makine öğrenmesi modellerinin performansı üzerindeki etkilerine ilişkin çalışmaların eksikliğini fark ettik ve bu konuyu ele alan birkaç çalışmaya rastladık [17, 20].

Last ve arkadaşları [18] NSVDist (Hassas Değer Dağılımları ile Homojen Olmayan Genelleme) adlı yeni bir anonimleştirme algoritmasını bilgi kaybı metriği ile tanıtmışlardır. Çalışmalarında 8 veri kümesi ve 4 ML modeli ile çalışmışlar, ayrıca algoritmalarını diğer üç algoritma olan Mondrian [21], PAIS [22] ve SeqA [23] ile karşılaştırmışlardır. Bizim çalışmamızla karşılaştırıldığında, ML için özel bilgi kaybı metriği ile otomatik bir genelleştirme çerçevesi sunuyoruz, üç k-anonimlik uzantısını karşılaştırıyoruz ve karşılaştırmak için daha fazla sınıflandırıcı kullanıyoruz. Bu çalışma, [19] ve bizim çalışmamız tarafından anonimleştirilmiş veri kümeleri için genel olarak kötü performans gösterdiği bildirilen Naive Bayes ve SVM modellerinin kullanımını göstermektedir.

Slijepcevic ve diğerleri [19] *k-anonimlik* için 4 farklı algoritmayı karşılaştırmıştır: Optimal Lattice Anonymization (OLA) [24], Mondrian [21], Top-Down Greedy Anonymization (TDG) [25] ve k-en yakın komşu kümeleme tabanlı (CB) algoritma [26]. Çalışmalarında 4 ML modeli ve 4 veri kümesi kullanmışlardır ve *k-anonimleştirmenin* sınıflandırma modelleri üzerindeki etkilerini kapsamlı bir şekilde incelemişlerdir. Bu çalışma bizim araştırma konumuza yakındır, ancak karşılaştırmak için, çalışmamızda k-anonimliğin farklı uzantılarını (yani l-çeşitlilik ve *t-yakınlık*) araştırdık, sadece bir tanesini değil. Ayrıca, makine öğrenimi için özelleşmiş bilgi metriğini kullanan yeni bir otomatik genelleştirme çerçevesi önermekte ve kullanmaktayız.

Fung ve arkadaşları [20] Mikro-birleştirme tabanlı Sınıflandırma Ağacı (MiCT) ile *k-anonimliği sağlayan* yeni bir *anonimleştirme* yöntemi sunmuşlardır. Yöntemlerini iki örnek üzerinde test etmişlerdir

veri kümeleri, standart Yetişkin veri kümesi ve Alman kredi veri kümesi. İki sınıflandırıcı kullanarak eğitim ve test yapmışlardır. Çalışmaları anonimlik elde etmek için farklı bir teknik (mikro toplama) kullanmasına rağmen, her iki çalışmanın da anonimleştirmenin sınıflandırıcı performansı üzerindeki etkisini araştırmayı amaçlaması açısından bizim çalışmamıza yakındır. Ancak, biz daha fazla *k-anonimlik* uzantısı ve daha fazla sınıflandırıcı üzerinde test ettik ve bu çalışma yaklaşımının, önerilen yöntemin davranışını anlamak için daha fazla sınıflandırıcı ve farklı dağılımlara sahip veri kümeleri ile de araştırılması gerektiğine inanıyoruz.

Wimmer ve arkadaşları [17] *k-anonimliğin* ML modellerinin performansı üzerindeki etkilerini araştırmayı amaçlamışlardır. Çalışmalarında 6 farklı ML modeli ve 3 farklı veri kümesi kullanmışlar, ML modellerinin parametrelerini kapsamlı bir şekilde incelemişler ve her model için doğruluk, F1 skoru ve karışıklık matrisi sonuçlarını raporlamışlardır. Ancak, kullanılan k-anonimlik algoritmasından bahsetmedikleri gibi genelleme ve bastırmanın nasıl kullanıldığından da bahsetmemişlerdir. Çalışmamızla karşılaştırıldığında, hem *k-anonimlik* algoritması tasarımı hem de bir dizi *k-anonimlik* uzantısı üzerinde genelleme ve bastırma kullanımı üzerinde kapsamlı bir şekilde çalışmakta ve raporlamaktayız. Ayrıca, modellerin performansını kullandıkları aynı metrikler açısından raporlamanın yanı sıra, ML sınıflandırıcıları için özelleştirilmiş genelleştirme çerçevesinde bir bilgi kaybı metriği kullandık.

1.2 Veri Gizliliği

Veri gizliliği, her bireyin kişisel bilgilerinin veya kendileriyle ilgili herhangi bir verinin diğer taraflarla nasıl ve ne zaman paylaşılacağını veya işleneceğini belirleyebilmesi gerektiği fikrini ifade eder. İnternet kullanımındaki artış, çeşitli amaçlarla tüm platformlarda veri toplanmasının artmasına yol açmıştır. Bu artış doğal olarak veri gizliliğine yönelik ilginin de artmasına yol açmıştır. Kişisel bilgilerin izinsiz kullanımı, aşağıdakiler gibi birçok istenmeyen sonuca yol açacaktır:

- Kişisel bilgileri toplayan kuruluşlar, kullanıcı onayı olmadan bunları üçüncü taraflara satabilir.
- Özel veriler kullanıcıları dolandırmak veya taciz etmek için kullanılabilir.
- Kişisel bilgiler, kişiler hakkında hassas bilgileri bilmek ve faaliyetlerini izlemek için kullanılabilir.

Kişisel bilgilerin bu şekilde istismar edilmesi bireyleri etkileyebilir ve tüm kuruluşları tehlikeye atabilir. Ortaya çıkan bu düzenleme ve kanunlar sayesinde birçok birey bu konuda daha bilinçli hale gelmiştir. Birçok çalışma bu konuyu ele aldı, ancak bu endişelerin birçok yönü hala ele alınmadığından daha fazla çalışmaya ihtiyaç var.

1.3 Anonimleştirme

1.3.1 *k*- anonimlik

k-anonimlik, veri kullanışlılığını korurken veri örneklerinin yeniden tanımlanamamasını sağlayan bir veri anonimleştirme tekniğidir. Elimizde *n* sayıda kayıt içeren bir veri kümesi olduğunu ve her kaydın *d* sayıda öznitelik içerdiğini varsayalım. Bu öznitelikler, yarı tanımlayıcılar ve hassas bir değer olmak üzere iki tür değer içerir. Bir grup yarı tanımlayıcı bir araya gelerek süper tanımlayıcılar oluşturur ve bu süper tanımlayıcılar bazen veri kaydını doğrudan tanımlayıcısına kadar izleyebilir. Örneğin, {Yaş, Cinsiyet, ZIP, Irk} özniteliklerinin bir kombinasyonu doğrudan bir kişiye bağlanabilir. Veri kümesinin ***k-anonimlik*** özelliğine sahip olması için her bir yarı tanımlayıcı çiftinin en az k kez ortaya çıkması, dolayısıyla her bir kaydın veri kümesindeki ***k-1*** diğer kayıttan ayırt edilemez olması gerekir. k-anonimlik özelliğini sağlamak için genellikle genelleştirme ve bastırma yöntemleri kullanılır. İlerleyen bölümlerde daha fazla ayrıntı verilecektir.

1.3.2 k-anonimlik Varyasyonlar

k-anonimlik, her bir veri grubunda, her bir kaydın k-1 diğer kayıttan ayırt edilemez olmasını garanti ederek elde edilir, bu da özne kimliğini şu anlamda korur

Bir düşman bir öznenin tüm quasi değerlerini bilse bile, düşman sadece öznenin hangi veri grubuna ait olduğunu bulabilecek, başka bir şey bulamayacaktır.

Sonraki alt bölümlerde bu mantığın ardındaki birkaç soruna değineceğiz ve önerilen çözümleri açıklayacağız.

***l-çeşitlilik*:** Standart *k-anonimlik için* önemli bir sorun, bir veri grubundaki tüm kayıtların aynı hassas özniteliği içerebilmesidir, bir kişinin belirli bir grupta olduğunu öğrenen bir düşman, hassas özniteliğin değerini kesin olarak bilebilecektir [9]. Bununla başa çıkmak için *k-anonimliğin* l-çeşitlilik [6] adı verilen bir varyasyonu önerilmiştir, *k-anonimliğe,* her veri grubunda en az I sayıda hassas öznitelik değeri içermesi gerektiğini belirten ekstra bir kriter ekler, bu akılda tutularak, bir düşman öznenin veri grubunu bulmayı başarsa bile, belirli bir özne için hassas özniteliği kesin olarak bulamayacaktır.

t-yakınlık: Daha önce tartıştığımız l-çeşitlilik varyasyonu ile ilgili bir sorun, bir saldırganın bir öznenin hassas özniteliğini bulmak için olasılıksal muhakeme kullanabilmesidir; bir veri grubunda 5 özneden 4'ü aynı hassas özniteliğe sahipse, saldırgan, hedeflenen öznenin aynı hassas özniteliğe sahip öznelerin çoğuna ait olduğuna dair bir kesinlik derecesine sahip olabilir [9]. t-yakınlık varyasyonu [7] bir adım daha ileri giderek, standart *k-anonimlik* için hassas özniteliklerin tüm veri kümesindeki istatistiksel dağılımının, *Kullback-Leibler* ıraksaması ile ölçülen her bir veri kaydındaki dağılımına makul ölçüde yakın olması gerektiğini belirten başka bir kriter ekler.

1.3.3 anonimleştirmeye yönelik saldırılar

Homojenlik [34]-[46]: Bir saldırgan, farklı kuruluşlardan yayınlanmış farklı veri kayıtlarını karşılaştırmak veya hatta öznenin yanında yaşamak gibi çok sayıda yöntemden Cinsiyet, Adres, Yaş, *iş alanı* gibi bir özne hakkında önceden bilgi sahibi olabilir

ve sosyal etkileşimler yoluyla bazı bilgiler elde etmek. Standart k-anonimlik ile ilgili bir sorun, anonimleştirme sürecinde bir araya getirilen bazı veri gruplarının aynı hassas değerleri içerebilmesidir. Saldırganın, kişinin ait olduğu bir kurumun (örneğin hastaneler veya okullar) anonimleştirilmiş ve yayınlanmış veri kayıtlarına erişimi olduğunu varsayarsak, saldırgan kişi hakkında bildiği Yaş, Adres, Etnik Köken ve Cinsiyet gibi bilgilerin bir kombinasyonunu kullanabilir ve kayıtları bu yarı tanımlayıcılara göre filtreleyebilir. Bunu yaparak saldırgan, aynı yarı tanımlayıcılara sahip ve büyük olasılıkla hepsi aynı hassas değeri paylaşan bir dizi kayıt elde eder, o zaman saldırgan, öznenin hassas değerini kesin olarak bilecektir.

Arka Plan Bilgisi Saldırısı [34]-[46]: Arka plan saldırıları ve Homojenlik saldırıları, saldırganın özne hakkında bazı yarı tanımlayıcılara erişimi olduğunu varsayması bakımından benzerdir. Bununla birlikte, Arka Plan saldırılarının gruplandırılmış veri kayıtlarının aynı hassas değeri paylaştığını varsaymaması, bunun yerine öznenin veri kayıtlarındaki hassas değerlerden birine sahip olma olasılığının statik olarak daha yüksek olduğunu bilmesi bakımından farklılık gösterirler. Bunun bir örneği, öznenin tıbbi verilerinin yayınlanması ve saldırganın cinsiyetini zaten biliyor olması ve hassas değerlerin kadınlarda erkeklerden daha sık görülen bir değer (örneğin bir hastalık) içermesidir. Bu tür bir bilgi, saldırganın veri kayıtlarını filtrelemesini ve hassas kayıtlar hakkında kesin bilgi edinmesini kolayca sağlayabilir.

1.4 Federated Learning

Federated learning [35] basit bir ifadeyle, makine öğrenimi modellerini eğitmek için merkezi olmayan bir yaklaşımdır ve tüm eğitim uç cihazlarda kullanıcı düzeyinde gerçekleşir. Verinin kendisi cihaz dışına aktarılmaz ve böylece aktarım sırasında veya veri merkezlerinde işleme veya eğitimin herhangi bir aşamasında oluşabilecek veri gizliliği riskleri azaltılır. Federal öğrenmenin temel versiyonunda global bir modele sahip merkezi bir sunucu mevcuttur. Eğitim için kullanılacak her bir uç cihazda yerel bir model bulunmaktadır. Yerel modeller uç cihazlardan gelen verileri eğitir ve eğitilen yerel model ağırlıklarını merkezi sunucuya aktarır ve global modeli tüm cihazların aldığı ağırlıklarla güncellemeye devam eder. Kenar cihazların

cihazların tamamen eğitilmiş global modeli kullanması gerekir, bu model cihazlarla paylaşılabilir, hiçbir ham veri cihazlarla veya sunucularla paylaşılmayacaktır. Bu yaklaşım 2016 yılında google tarafından tanıtıldı, ancak o zamandan bu yana veri gizliliği ve Federated learning'in daha verimli hale getirilmesiyle ilgili birçok zorluğu ele almak için birçok geliştirme üzerinde çalışıldı ve uygulandı. Temel yapısal teori hala aynıdır ve bunu aşağıda daha ayrıntılı olarak açıklayacağız:

- Yerel model eğitimini gerçekleştirmek için bir dizi Edge cihazı (istemci) seçilir. Seçim kriterleri değişiklik gösterir, mevcut en iyi istemcileri seçmek için kullanılan birden fazla algoritma ve teknik vardır.
- Global model ağırlıkları değişimi, müşterilerin mevcut Global modele zaten toplanmış olan önceki bir oturumdan başlayarak yerel modellerini eğitmeye başlamaları gerektiği durumda. Bunu bir başlangıç noktası olarak düşünebiliriz.
- Müşterilerin cihazlarında yerel model eğitimi. Yerel modellerin, her cihazın yerel olarak sahip olduğu veya yerel olarak topladığı ve küresel modelin amaçladığı görevle ilgili verilerle eğitilmesi.
- Yerel Modeller, eğitimden sonra yeni oluşturdukları ağırlıkları global model ile değiştirir. Daha sonra global model bu ağırlıklardan güncellenecektir.
- Süreç, istenen hedefe ulaşılana kadar tekrarlanabilir.

1.4.1 Federasyon Türleri Öğrenme

Veri dağıtımı ile ilgili türler:

Yatay FL [37],[38]: Bu tür, her müşterinin cihazının eğitim için kullanılacak verilerde aynı özellik setine sahip olduğu, ancak farklı kayıtlara sahip olduğu durumlarda kullanılır. Yatay FL'ye bir örnek, birden fazla hastanenin aynı tür tıbbi verileri toplaması, ancak her veri kaydının farklı bir konuyu temsil etmesidir.

Dikey FL [37],[38]: Bu tür, her istemci cihazın eğitimde kullanılacak farklı bir veri özelliğine sahip olduğu durumlarda kullanılır. Dikey FL'ye bir örnek, hastanelerin tıbbi geçmişleriyle ilgili bir dizi konu için bir özellik veri kümesine sahip olması ve bir dış

laboratuvarında aynı denekler için kan testleri sonuçlarıyla ilgili başka bir veri özellikleri kümesi vardır.

Müşteri Seçimi ile ilgili türler:

Çapraz silo FL [36], [38]-[40]: Bu tür federe öğrenme, sınırlı sayıda istemci olduğunda ancak tüm eğitim turları için mevcut olduğunda kullanılır, veri dağıtımı yatay veya dikey olabilir ve bu tür FL genellikle Hastaneler ve kuruluşlar tarafından kullanılır.

Cihazlar Arası FL [36], [38]: Bu tür federe öğrenme, çok sayıda istemci olduğunda kullanılır, ancak hepsi güvenilir değildir, seçim algoritmaları ve teknikleri bu türde yoğun olarak kullanılır, istemcinin eğitimi kesintiye uğratması gibi başka sorunlar da ortaya çıkabilir, bu nedenle bu tür genellikle eğitim süreci boyunca dikkatli bir şekilde izlenir.

Tamamen Merkezi Olmayan Federe Öğrenme

Normal federe öğrenmede, geniş bir dizi istemci ve uç cihazdan gelen tüm ağırlıkları ve katkı alışverişini idare etmek için hala merkezi bir sunucuya ihtiyaç vardır, bu nedenle doğal olarak böyle bir merkezi sunucu FL'deki en önemli bileşendir. Potansiyel olarak bu yaklaşımın zayıf noktası, herhangi bir noktada çökmesi veya başarısız olması durumunda ciddi sonuçlar doğuracak olmasıdır.

Bu soruna bir çözüm, merkezi bir sunucunun olmadığı ve küresel modelin durumunun artık mevcut olmadığı eşler arası dağıtılmış öğrenmedir [36], [41]. Çalışma şekli, her istemcinin iletişim kuracağı küçük bir komşu istemci kümesine sahip olmasıdır, her istemci eğitim sürecinde yerel bir güncelleme sağladığında ve bunları komşu istemcileriyle paylaştığında her eğitim turu başarılı kabul edilir. Bu süreç sayesinde istemciler tüm ağ boyunca kabaca aynı yerel model ağırlıklarına sahip olurlar.

1.4.2 Federated Learning'e Saldırılar

Gizlilik ve Güvenlikle İlgili Saldırılar

Üyelik Çıkarım Saldırıları: ML ve FL uygulamaları, saldırganın eğitim verilerine ve hatta çoğu durumda modellerin parametrelerine veya ağırlıklarına erişimi olmadığı durumlarda eğitim sürecinde kullanılan veri kayıtlarının tespit edilmesini mümkün kılan Üyelik Çıkarım Saldırılarına [42] karşı savunmasızdır. Bu tür saldırılar, ML ve FL Modellerinin daha önce gördüğü (eğitim sürecinde) veri kayıtları ile daha önce hiç görmediği diğer kayıtlar üzerinde farklı davrandığı gerçeğinden faydalanır. Saldırgan bu gözlemleri elde ettiğinde, hangi veri kayıtlarının eğitim sürecinin bir parçası olduğunu ve hangilerinin olmadığını belirleyebilir. Olası bir senaryoya göre, bir uygulama FL'yi herkese açık olarak sunulan bir hizmette kullanıyorsa, saldırgan hizmeti kullanabilir ve kendi veri kaydını sağlayabilir ve ardından hizmet sonuçlarını (FL model çıktısı) gözlemleyerek model ve veri kayıtları hakkında bilgi edinebilir. Bu tür saldırılarla başa çıkmak için veri şifreleme ve şifrelenmiş veriler üzerinde eğitim, veri anonimleştirme ve diferansiyel gizlilik yöntemleri gibi birçok çözüm önerilmiştir. Bunlardan ilki, özellikle büyük miktarda veriyi eğitmemiz gerektiğinde hesaplama ek yükü getirip çok zaman alırken, son ikisi verilere modellerin performansını etkileyebilecek gürültü ekler ve kullanılabilirlik ile gizlilik arasında bir uzlaşma olarak kabul edilir.

Veri ve Model Zehirlenmesi Saldırıları: Veri zehirleme saldırıları [38], model davranışını bozmak veya etkilemek amacıyla bir veya daha fazla istemci cihazdaki eğitim verilerini kirletmeyi ve bunlara veri kayıtları eklemeyi içerir. Daha sonra bu eğitilmiş yerel modeller, küresel modu güncellemek için merkezi sunucuya gider. Saldırganın eğitim sürecini ve model çıktılarını etkilemeyi amaçlaması durumunda bu saldırı hedefli zehirleme saldırıları grubuna, saldırganın yalnızca model tahmin sonuçlarını yanlış kabul etmek için yanlış veri eklemeyi amaçlaması durumunda ise hedefli olmayan saldırılar grubuna girer. Bu saldırılar, modellerin eğitim için müşterilerin verilerine dayanıp dayanmadığına bağlıdır. Model zehirleme saldırıları [38] veri zehirleme saldırılarına çok benzemektedir ancak aralarındaki fark eğitim verilerini bozmak yerine yerel modellerin parametrelerini bozarak global modele güncellenecek hatalar eklemeyi amaçlamalarıdır. Orada

Bu tür saldırılarla başa çıkmanın kolay bir yolu yoktur, özellikle de global model güncellendikten sonra tespit edilmişse, bu durumda yerel model güncellemelerinin kapsamlı bir analizi gerçekleşecek ve özellikle bir saldırgan farklı eğitim döngüleri içinde verileri veya modelleri zehirlediğinde bozuk güncellemeleri tespit etmek çok zaman ve kaynak alacaktır. Bu saldırıları önlemek için, bozuk istemci cihazlarının tespiti, güncellemeleri kullanmadan önce yapılmalıdır. Bu, istemci seçim stratejileri ve algoritmaları, istemci doğrulaması ve anomali tespiti yoluyla gerçekleştirilebilir.

Eğitimle İlgili Dezavantajlar

Bağlantılar ve İletişim: Tasarım gereği FL, tüm istemciler global modeli güncellemesi için yerel modellerinin ağırlıklarını merkezi sunucu ile paylaşacağından dikkatlice optimize edilmiş bir iletişim stratejisi gerektirir. İstemci cihazların bağlantıları genellikle ağ içi aktarım hızlarına göre değişir ve neredeyse her zaman merkezi sunucuların bağlantı hızlarından daha yavaştır. Bu durum FL için büyük bir darboğaz ve ciddi ve pahalı bir zorluk yaratabilir. Bunun çözümü, yerel modellerin güncellemeleri için iletişim bant genişliğini en aza indirerek ele alınabilir, bu özel konunun önemi nedeniyle doğal olarak çok dikkat çekmiş ve birçok çalışma için ilgi konusu olmuştur.

1.5 Bu Çalışmasının Arkasındaki Motivasyon

Veri gizliliği birçok birey ve kuruluş için giderek artan bir endişe kaynağıdır. Doğal olarak birçok ülkede veri gizliliğini korumaya yönelik düzenlemeler ortaya çıkmış ve uygulanmaktadır. Bu tür düzenlemeler, kuruluşların veri toplama ve kullanma becerileri için bir zorluk oluşturmaktadır. Bu zorluklarla yüzleşmek için veri anonimleştirme ortaya çıktı ve birçok veri odaklı kuruluş bu yaklaşımı benimsedi. Yakın zamanda ortaya çıkan bir diğer yaklaşım ise, kullanıcıların ham verilerini toplamadan veya paylaşmadan makine öğrenimi modellerinin merkezi olmayan bir şekilde eğitilmesine olanak tanıyan FL'dir. Bu çalışmada, k-anonimleştirilmiş verileri kullanmaya adapte olabilen bir çapraz-silo federasyon öğrenme çerçevesi tasarladık. Veri anonimleştirmenin entegrasyonunun daha iyi gizlilik sağlarken minimum bilgi kaybı sağlayabileceğini ve her iki yaklaşımın tek bir çerçevede kullanılmasının bize olanak sağladığını gösteriyoruz

Her iki yaklaşımın avantajlarından faydalanmak için. Bu çalışma konusu, yeni gelişmelere rağmen veri gizliliğinin hala tehlikede olabileceği ve bunu önlemek için seçenekleri araştırmaya devam etmemiz gerektiği gerçeğine dayanarak yapılmıştır. Veri anonimleştirme yaklaşımımız, verilerin gizliliğini ve işlenirken kaynakların en iyi şekilde kullanılmasını garanti edecektir. Bu çalışma, veri gizliliği yöntemlerinin makine öğrenmesi algoritmalarının performansı üzerindeki hesaplama veya çıktı kalitesi açısından dezavantajlarının, IoT'de uygun veri gizliliği teknikleri ve makine öğrenmesi algoritmalarının seçimi kullanılarak en aza indirilebileceği hipotezini değerlendirmek için tasarlanmıştır.

1.6 Tez Yapısı

Tez aşağıdaki şekilde yapılandırılmıştır:

- Bu tezin ilk bölümünde anonimleştirmenin arkasındaki mantık ve anonimleştirme çerçeve kurulumumuz ele alınmaktadır. ***K-anonimleştirmenin ve*** bu çalışma boyunca kullanacağımız diğer varyasyonların birkaç kilit noktasını ve önemli bileşenlerini ele alıyoruz.
- Bölüm 2'de, bu çalışmada kullandığımız, verilere anonimleştirme uygulandıktan sonra bilgi kaybını tahmin etmek için kullanılan entropi katsayısına dayalı bir bilgi kaybı metriğini açıklıyoruz.
- Bölüm 3'te birlikte çalıştığımız makine öğrenimi modellerini ve sınıflandırıcıları tartışıyor, her modelin arkasındaki teorik mantığı açıklıyor ve bu çalışmada kullandığımız değerlendirme metriklerinden ayrıntılı olarak bahsediyoruz.
- Bölüm 4'te ***k-anonimleştirme***, l-çeşitlilik ve t-yakınlık varyasyonlarımızı birleştirdiğimiz ***anonimleştirme*** çerçevesi nihai tasarımımızdan bahsettik ve bunları verilere nasıl tanıttığımızı ve modelleri eğitmek ve değerlendirmek için nasıl kullandığımızı açıkladık,
- Bölüm 5'te anonimleştirme sonuçlarımızı, her bir veri kümesi için bilgi kaybı sonuçlarımızı ve anonimleştirme varyasyonlarını raporluyoruz. Ayrıca sınıflandırıcı sonuçlarımızı daha önce açıklanan metriklerle gösteriyoruz, bu bölümü hem modeller hem de anonimleştirme performansı hakkında birkaç açıklama ile bitiriyoruz.

- Bölüm 6'da, her bir yönü ayrıntılı olarak ele aldığımız FL Çerçevemizden bahsediyoruz, istemciler arasında veri dağıtımını, ağırlık toplama modellerini ve maksimum gizlilik elde etmek için anonimleştirmenin federe öğrenmeye entegrasyonunu tartışıyoruz.
- Bölüm 7'de, anonimleştirme işleminden sonra bilgi kaybı açısından FL çerçeve sonuçlarımızı ve eğitim sürecinde bir dizi istemci kullanan küresel modelin son versiyonundaki sınıflandırma sonuçlarını rapor ediyoruz.
- Bölüm 8'de, çalışmamızı ve nihai bulgularımızı açıklayarak bu çalışmayı sonuçlandırıyoruz ve bu alan ve çalışma için gelecekteki yönleri öneriyoruz.

2. K-ANONIMLIK KURULUMU

2.1 Mondrian *k*- anonimlik

Mondrian [21], k-anonimlik elde etmek için kullanılan katı çok boyutlu bölümleme için yukarıdan aşağıya açgözlü bir yaklaşım algoritmasıdır. Mondrian algoritması aşağıdaki gibi çalışır. Elimizde nihai bölümleme kümesi *F ve* başlangıçta tüm veri kümesini bir bölüm olarak içeren bir çalışma kümesi *S olduğunu* varsayalım. Daha sonra, bölümdeki boyutların (sütunların) göreceli açıklıklarını hesaplar ve sıralarız. Her boyut için, bu durumda medyan olan bir bölme noktasına dayalı olarak sütun uzayı boyunca bölümü böleriz. Bundan sonra, yeni bölümün k-anonimlik kriterlerini karşılayıp karşılamadığını kontrol ederiz. Eğer öyleyse, ortaya çıkan bölümleri *S* kümesine ekleriz ve yeni bölümler üzerinde aynı önceki işlemi yineleriz. Geçerli bir bölünme elde edilememişse ve sütunlar uzayında hiçbir bölge kalmamışsa, bölümü *F kümesine* ekleriz. Tüm bunlar tamamlandığında, yaklaşık *k-anonimleştirilmiş* bir veri kümesi elde ederiz.

Bu çalışmada Mondrian algoritmasının k-anonimlik ve diğer uzantıları olan *l-çeşitlilik* ve *t-yakınlık* elde etme kabiliyetini araştırıyoruz. Mondrian algoritmasının uygulanmasında, bölümlemenin dayanması gereken boyutların seçilmesine de izin verdik. Bu, yarı-kimliklendiricilerimiz olmadığında ve bunları anonimleştirme sürecine dahil etmek istemediğimizde önemlidir. Ayrıca, Genelleştirme ve Bastırma bölümüne bilgi kaybının daha iyi azaltılmasını sağlayan bir optimizasyon adımı entegre ettik.

2.2. Kayıp metriği

Bir anonimleştirme yaklaşımının (önerilen genelleştirme ve bastırma yöntemleri dahil) kalitesini değerlendirmek ve bilgi kaybının ML modellerinin performansını nasıl etkilediğini ölçmek için, anonimleştirme işlemi nedeniyle kaybedilen verileri ölçmemiz gerekir. X ve Y'nin iki rastgele değişken olduğu durumlarda, Y'yi kullanarak X'in ne kadarını tahmin edebileceğimizi söyleyen bir ilişki ölçüsü olan entropi katsayısını [5], [10] kullanmaya karar verdik.

k-anonimlik yaklaşımımızın (önerilen genelleştirme ve bastırma yöntemleri dahil) kalitesini değerlendirmek ve bilgi kaybının ML modellerinin performansını nasıl etkilediğini ölçmek için, anonimleştirme işlemi nedeniyle kaybedilen verileri ölçmemiz gerekir.

Koşullu entropiyi hesaplayarak başlıyoruz, burada P(x,y):

$$H(X \mid Y) = -\sum_{x,y} P(x,y)(x \mid y) \log P(x \mid y)$$

Daha sonra Entropi katsayısını hesaplamaya geçiyoruz:

$$U(X \mid Y) = \frac{H(X) - H(X \mid Y)}{H(X)} = \frac{I(X;Y)}{H(X)}$$

Sonunda 0 ile 1 arasında değişen bir ilişkilendirme değeri elde edeceğiz; burada 1 tam ilişkilendirme (veri kaybı yok) ve 0 ilişkilendirme yok (tam veri kaybı) anlamına gelir.

Veri kümelerimizi makine öğrenimi modellerinde kullanmayı hedeflediğimizden, gizliliği mümkün olduğunca korurken veri kaybını en aza indirmeye odaklandık. Bu metriği esas olarak Etki Alanı Değişkenleri ile Hedef Değişken arasındaki ilişkiyi ölçmek için kullanacağız.

2.3 Optimizasyon Adım

Anonimleştirme sürecinde, her bir bölünmüş kayıt grubu içinde Mondrian algoritması çıktılarına daha iyi yaklaşmaya çalışan bir optimizasyon adımı sunuyoruz. Tüm Mondrian bölünmüş kayıtlarını içeren bir *R Kayıt* kümesine sahip olduğumuzu varsayalım. Her kayıt $i \in R$

L_i satır kümesine sahiptir ve her satır bir dizi öznitelik değerine (yarı tanımlayıcılar) sahiptir. Her L_i kümesi için aşağıdakileri yaparız:

1. Tam olarak aynı değer dizisine sahip en az k satır olup olmadığını kontrol edin, dolayısıyla k kriterini karşılar. Bu satırları $E_i \in L_i$ alt kümesine ekleriz.

2. *k* kriterini sağlayan değerler dizisi için yaklaşık bir eşleşmeye sahip olan satırları kontrol edin. Bu satırları $A_i \in L_i$ alt kümesine ekleriz.

Sonunda, *k* kriterini karşıladıkları ve tam eşleşme oldukları için daha fazla işlem gerektirmeyen E_i alt kümesi ve Genelleştirme ve Bastırma adımına iletilecek olan A_i alt kümesi elde edilir.

2.4 Genelleme ve Bastırma

Mondrian, veri kümelerini *k*- anonimleştirilmiş kayıtlara bölmeyi amaçlayan açgözlü bir yaklaşım algoritması olduğundan, algoritmanın tam olarak uygulayamadığı kayıtlar üzerinde *k*- anonimlik kriterini uygulamak için Genelleştirme ve Bastırma yöntemlerini kullanmalıyız. Genelleştirme, her bir alanın bireysel değerlerini daha az spesifik daha geniş bir değerle değiştirme işlemidir; örneğin, 18, 19, 21, 25, 31 ve 38 değerlerine sahip Yaş sütununa sahip bir veri kümemiz varsa, bunları <20, 20-30 ve 30-40 olarak genelleştirebiliriz. Bastırma, belirli bir satırdaki değerin tamamen kaybolduğunu belirtmek için belirli alan değerlerini kısmen veya tamamen yıldız işareti '*' ile değiştirme işlemi iken, örneğin 114554 , 114553 ve 114543 değerlerine sahip ZIP sütununa sahip bir veri kümemiz varsa, bunları kısmen 1145**, 1145** ve 1145** olarak bastırabiliriz. Tablo 2.1'de Yaş Sütununda genelleştirme ve Uyruk sütununda Bastırma gösterilmektedir.

Tablo 2.1: Genelleştirme ve bastırma ile anonim hale getirilmiş veri kayıtları

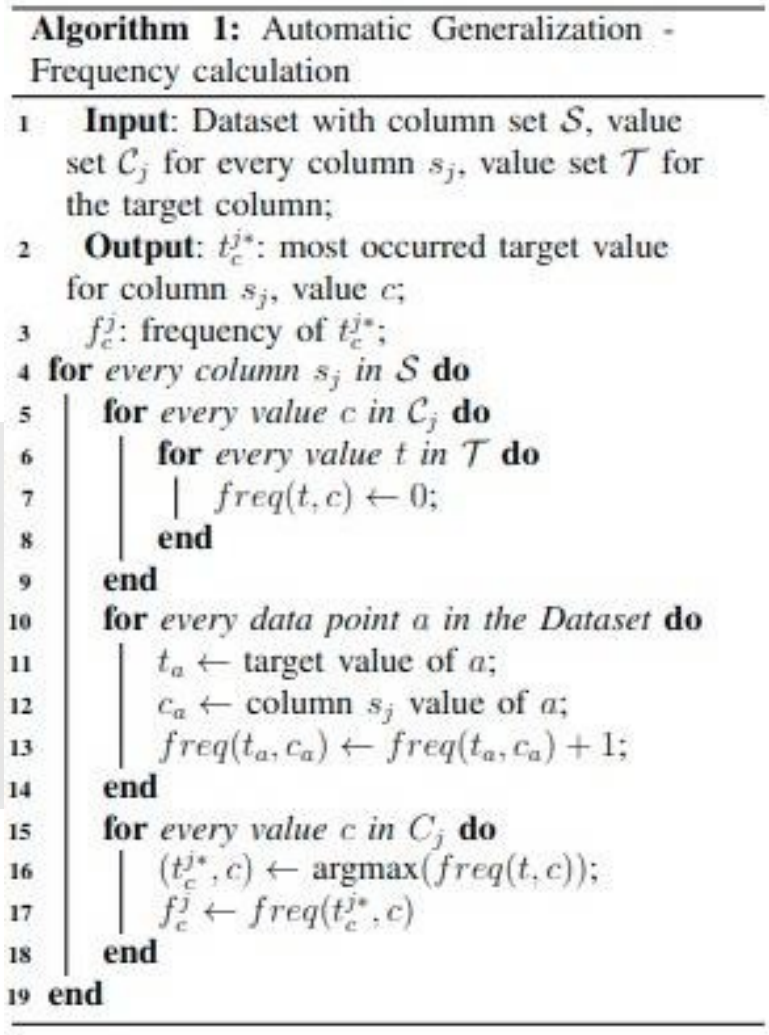
#	**ZIP**	**Yaş**	**Uyruk**	**Hastalık**
1	130**	< 25	*	Grip
2	130**	< 25	*	Kalp Hastalığı
3	130**	< 25	*	Kanser

Bu çalışmada, bastırma stratejisi de dahil olmak üzere genelleştirme kullanıyoruz. Hem genelleştirme hem de bastırma yöntemleri k-anonimlik kriterini uygulamak için kullanılmaktadır. Anonimliği zorlarken veri bozulmasını azaltmayı hedefliyoruz. Bunu başarmak için hem minimum bastırma politikasını hem de [4]'te tanımlanan minimum göreli uzaklık politikasını benimsiyoruz. Başka bir deyişle, genelleştirme yaklaşımımız daha az bastıran ve hiyerarşinin yüksekliğine göre toplam göreli adım sayısını en a z a i n d i r e n genelleştirmeyi tercih eder.

Bir sonraki alt bölümde, hem kategorik hem de sayısal veriler için otomatik olarak kurallar oluşturabilen yeni bir Genelleştirme yöntemi öneriyoruz ve strateji hedeflerini garanti altına almak için önceki alt bölümde açıklandığı gibi ek bir optimizasyon adımı entegre edilecektir.

2.5 Otomatik Genelleştirme

Genelleştirme, *k* kriterini uygulamak için çok güçlü bir yöntemdir ve çoğu durumda kayıt genelleştirme hiyerarşilerine manuel olarak karar vermeyi içeren bildirimsel genelleştirme kullanılır. Manuel olarak oluşturulan hiyerarşiler daha çok dışa bağlı mantıkla ilişkilidir, aykırı değerlerin genellikle tamamen hariç tutulması nedeniyle bazen verileri çarpıtabilir veya önyargılı olabilir, ayrıca kayıt değerlerinin hedef niteliklerle korelasyonunu tamamen göz ardı eder. Biz, kayıt değerlerinin hedef özniteliklerle korelasyon frekanslarına dayanan otomatik toplama tabanlı bir genelleştirme hiyerarşisi öneriyoruz. Önerilen otomatik genelleştirme algoritması iki ana adımdan oluşmaktadır. İlk adım, Şekil 2.2'de gösterildiği gibi sütun değeri-hedef değer çiftleri başına frekans hesaplamasıdır. İkinci adım ise Şekil 2.3'te gösterilen frekans tabanlı gruplamadır.



Algorithm 1: Automatic Generalization - Frequency calculation

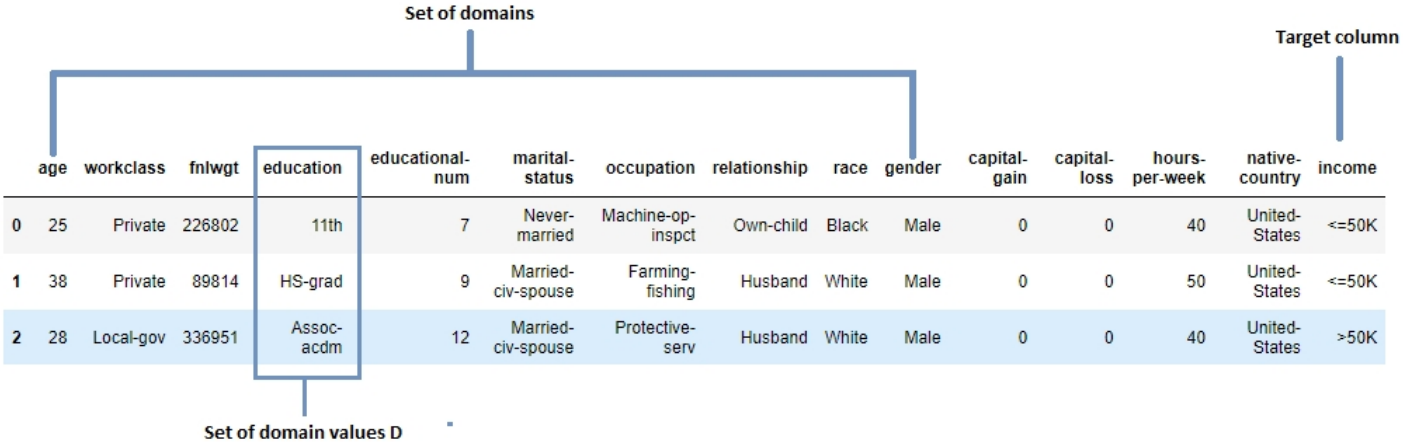
Input: Dataset with column set $\mathcal{S}$, value set $\mathcal{C}_j$ for every column s_j, value set $\mathcal{T}$ for the target column;
Output: t_c^{j*}: most occurred target value for column s_j, value c;
f_c^j: frequency of t_c^{j*};
for *every column* s_j *in* $\mathcal{S}$ **do**
  for *every value* c *in* $\mathcal{C}_j$ **do**
    for *every value* t *in* $\mathcal{T}$ **do**
      $freq(t, c) \leftarrow 0$;
    end
  end
  for *every data point* a *in the Dataset* **do**
    $t_a \leftarrow$ target value of a;
    $c_a \leftarrow$ column s_j value of a;
    $freq(t_a, c_a) \leftarrow freq(t_a, c_a) + 1$;
  end
  for *every value* c *in* C_j **do**
    $(t_c^{j*}, c) \leftarrow \text{argmax}(freq(t, c))$;
    $f_c^j \leftarrow freq(t_c^{j*}, c)$
  end
end

Şekil 2.1: Otomatik Genelleştirme Algoritması 1 - Frekans Hesaplamaları

Algorithm 2: Automatic Generalization - Frequency Based Grouping

Input: (t_c^{j*}, f_c^j) for every value c of every column s_j, ρ: percentile range value;
Output: $Group[j][groupid]$, an array of set of values for every column s_j;
for *every column* s_j *in* $\mathcal{S}$ **do**
  for *every target* t *in* $\mathcal{T}$ **do**
    for *range: 1 to* $100/\rho$ **do**
      $ColumnSet[t][range] \leftarrow \emptyset$;
    end
  end
  for *every value* c *in* C_j **do**
    $ColumnSet[t_c^{j*}][\lfloor f_c^j/\rho \rfloor] \leftarrow ColumnSet[t_c^{j*}][\lfloor f_c^j/\rho \rfloor] \cup c$;
  end
  $groupid \leftarrow 0$;
  for *every target* t *in* $\mathcal{T}$ **do**
    for *range: 1 to* $100/\rho$ **do**
      if $ColumnSet[t][range] \neq \emptyset$ **then**
        $groupid \leftarrow groupid + 1$;
        $Group[j][groupid] \leftarrow Columnset[t][range]$;
      end
    end
  end
end

Şekil 2.2: Otomatik Genelleştirme Algoritması 2 - Frekans Tabanlı Gruplama



	age	workclass	fnlwgt	education	educational-num	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	native-country	income
0	25	Private	226802	11th	7	Never-married	Machine-op-inspct	Own-child	Black	Male	0	0	40	United-States	<=50K
1	38	Private	89814	HS-grad	9	Married-civ-spouse	Farming-fishing	Husband	White	Male	0	0	50	United-States	<=50K
2	28	Local-gov	336951	Assoc-acdm	12	Married-civ-spouse	Protective-serv	Husband	White	Male	0	0	40	United-States	>50K

Şekil 2.3: Otomatik genelleştirmede kullanılan örnek veri kayıtları.

C sütun kümesine sahip bir veri setimiz olduğunu varsayalım $\{s_1, s_2, \ldots, s_c\} \in S$, her s_j sütunu bir n_j benzersiz değerler kümesine sahiptir $\{c_1^j, c_2^j, \ldots, c_{nj}^j\} \in C_j$. Ayrıca m *adet* benzersiz değerden oluşan bir kümeye sahibiz $\{t_1, t_2, \ldots, t_m\} \in T$ Hedef sütununda.

Her $s_j \in S$ sütunu için, C_j adresindeki her bir değeri T'deki her bir değerle birleştirerek başlarız, yani tüm çiftleri $(c_i^j, t_k), \forall i \in \{1,2,\ldots,n\}, \forall k \in \{1,2,\ldots,m\}$. Daha sonra, hesaplarız tüm veri kümesi üzerinde birleştirilmiş çiftlerin sıklığı (yani iki değerin kaç kez aynı satırda göründü). Her s_j sütunu ve c_i^j değeri için (c_i^j, t_k) çiftini seçiyoruz. en yüksek frekans. c_i^j değeri için bu en yüksek frekans f_i^j ile gösterilir ve karşılık gelen hedef değer t_i^* ile gösterilir.

Yetişkin veri setindeki "eğitim" sütununu ele alalım. Hedef sütununda (gelir) iki değer bulunmaktadır. Eğitim sütunundaki ve hedef sütunundaki her değer çifti için frekansları hesaplıyoruz ve en yüksek frekanslara sahip çiftler Tablo 2.2'de gösterilmektedir.

Tablo 2.2: Veri kümesindeki hedef sütunlarla birlikte en sık rastlanan eğitim sütunu benzersiz değerleri.

Education	Income	Frequency Percentage
Prof-school	>50K	0.7398
Doctorate	>50K	0.7255
Masters	>50K	0.5491
Preschool	<=50K	0.9879
11th	<=50K	0.9492
1st-4th	<=50K	0.9676
5th-6th	<=50K	0.9469
9th	<=50K	0.9457
10th	<=50K	0.9373
7th-8th	<=50K	0.9350
12th	<=50K	0.9269
HS-grad	<=50K	0.8414
Some-college	<=50K	0.8103
Assoc-voc	<=50K	0.7467
Assoc-acdm	<=50K	0.7420
Bachelors	<=50K	0.5871

Ardından, ikinci adım olan frekans tabanlı gruplama ile devam ediyoruz. İlk olarak, 100/ ρ bir tamsayı olacak şekilde bir yüzdelik aralık değeri ρ belirliyoruz. Daha sonra tüm aralığı böleriz
0 ile 100 arasında eşdeğer büyüklüklere sahip $100/\rho$ yüzdelik aralıklara, yani (0 - ρ, ρ - 2ρ, ... , 100 - ρ - 100). Örnek olarak, eğer $\rho = 10$ ise, o zaman 10 yüzdelik aralık olacaktır ve eğer $\rho = 20$, 5 aralık olacaktır. Her yüzdelik aralık farklı bir gruba karşılık gelir. Daha sonra her sütun değerini c^j en çok karşılaşılan hedef değerine ve f^j frekansının karşılık gelen yüzdelik aralığına göre gruplandırırız. Daha sonra sütunu genelleştiririz
değerleri bu temsili gruplara ayırmıştır.

Şimdi, Şekil 2.3'te gösterilen yetişkin veri setindeki "eğitim" sütununu tekrar ele alalım ve ρ değerini 10 olarak belirleyelim. Tablo 2.2'de gösterilen frekans değerleri, eğitim sütunu değerlerinin verilen yüzdelik dilim ile aşağıdaki altı gruba dahil edileceğini göstermektedir aralıklar.

- Grup 1 = (lise mezunu, biraz üniversite)

- Grup 2 = (Assoc-acdm, Assoc-voc)
- Grup 3 = (Prof-okul, Doktora)
- Grup 4 = (Lisans)
- Grup 5 = (Master)
- Grup 6 = (11., 10., 7.-8., 5.-6., 9., 12., 1.-4., okul öncesi)

Bu yöntemin 16 değeri otomatik olarak 6 temsili gruba genelleyebildiğini fark edebiliriz ki bu oldukça etkileyicidir. ρ değeri ne kadar küçük olursa o kadar iyi olduğu açıktır, bir gruptaki değerler ne kadar benzer olursa ve çok düşük ρ değeri (2 gibi) ile çoğunlukla hiç genelleme yapılmamasına neden olur. Öte yandan, koşul değeri ne kadar büyük olursa, o kadar az grubumuz olacak ve daha fazla genelleme gerçekleşecektir. Doğal bir soru bu noktada en iyi ρ değerine nasıl karar verileceğidir. Buradaki cevap farklı ρ değerlerini denemektir ve her bir değer için bir hiyerarşi oluşturuyoruz. Ardından, anonimleştirme işlemi sırasında genelleştirme ağaçlarını veri kayıtlarına uyguluyoruz ve her ağaç için önerdiğimiz metrik ile bilgi kaybını ölçüyoruz.

Verilen yüzdelik aralık koşuluyla bu yaklaşım aşağıdaki grupları üretecektir: Bu yöntemin 16 değeri otomatik olarak 6 temsili gruba genelleyebildiğini fark edebiliriz ki bu etkileyicidir. Açıkça görülmektedir ki ρ değeri ne kadar küçükse bir gruptaki değerler daha benzer olacaktır ve çok düşük ρ değeri (2 gibi) ile çoğunlukla hiç genelleme yapılmamasına neden olur. Öte yandan, koşul değeri ne kadar büyük olursa, o kadar az grubumuz olacak ve daha fazla genelleme gerçekleşecektir.

Bu noktada doğal bir soru, en iyi ρ değerine nasıl karar verileceğidir. Buradaki cevap, farklı ρ değerlerini denemek ve her değer için bir hiyerarşi oluşturmaktır. Ardından, anonimleştirme sırasında sürecinde genelleştirme ağaçlarını veri kayıtlarına uyguluyoruz ve her ağaç için önerdiğimiz metrikle bilgi kaybını ölçüyoruz. Her sütun için en iyi genelleştirme ağacını seçerek, yani en iyi genelleştirme ağacını sunan ρ değerini seçerek bilgi kaybını en aza indirmeye çalışıyoruz.

en az kayıp ve k kriterini karşılar. için farklı bir ρ değeri bulabileceğimizi unutmayın. farklı sütunlar.

Bu çalışmada Değer Genelleştirme Hiyerarşisi (VGH) üzerine odaklanmamıza rağmen, ZIP numaraları gibi alanlar için kullanılabilecek Alan Genelleştirme Hiyerarşisi (DGH) için de seçenek sunuyoruz.

Şekil 2.2'deki Algoritmanın zaman karmaşıklığı O(*cn*)'dir; burada n, veri kümesindeki veri noktalarının sayısıdır. Sütun sayısı (*c*) tipik olarak küçük bir sayı olduğundan, bu karmaşıklık doğrusal olarak kabul edilebilir. Şekil 2.3'teki Algoritmanın zaman karmaşıklığı daha da düşüktür, çünkü sütun sayısı, sütun başına değer sayısı, hedef değer sayısı ve aralık değer sayısının bir fonksiyonudur ve hepsi veri kümesinin boyutundan bağımsızdır.

3. MAKINE ÖĞRENME

3.1 Makine Öğrenimi Modelleri

Bu çalışmada, anonimleştirme sürecinin Makine Öğrenimi üzerindeki etkilerini anlamak için bir dizi makine öğrenimi modeli ile çalıştık.

Rastgele Orman: Karar ağaçları ve torbalama tekniklerini kullanan bir topluluk tekniğidir [27], verilerin rastgele seçilmiş bir alt kümesinden (eğitim seti) bir dizi karar ağacı oluşturarak başlar ve nihai sonucun belirlenmesinde yer almak için sonuçlarını toplar.

Son Derece Rastgele Ağaçlar (Extra Trees) Sınıflandırıcısı: Çeşitli karar ağaçları sonuçlarını bir araya getiren topluluk öğrenme tekniklerine aittir [28]. Rastgele ormana çok benzer ve sadece karar ağaçlarını oluşturma yönteminde farklılık gösterir.

Destek Vektör Makineleri (SVM): Esas olarak sınıflandırma için kullanılan iyi bilinen bir denetimli öğrenme algoritmasıdır. Karar vermeye yardımcı olmak için verileri en iyi şekilde ayıran optimum karar sınırlarını (hiper düzlem) bularak çalışır [29].

Gradyan Güçlendirme: Eğitim sürecinde güçlendirmeyi kullanmasıyla bilinen bir topluluk tekniğidir. İlk modelin eğitim verileriyle oluşturulduğu, ardından devam eden modellerin selefinin artık hatasıyla eğitildiği bir dizi model oluşturarak çalışır [30].

Yapay Sinir Ağı (YSA): Bu çalışmada, eğitim sürecinde ileri besleme ve geri yayılım kullanan çok katmanlı bir algı sınıflandırma algoritması kullandık [31]. Ayrıca parametre optimizasyonu için Grid Search algoritmasını kullandık.

Bu algoritmalara ek olarak, **K-Nearest Neigbour (KNN), Logistic Regression, Gaussian Naive Bayes (GaussianNB)** ve **Stochastic Gradient Descent (SGD) sınıflandırıcı** gibi diğer ML modellerini de inceliyoruz.

3.2 Makine Öğrenimi Değerlendirmesi Metrikler

Son aşamada, sonuçlarımızı değerlendirmek için farklı performans metriklerini karşılaştırıyoruz. Sınıflandırma Doğruluğu, Kesinlik Puanı, Geri Çağırma ve F1 puanlarını elde ediyoruz.

Sınıflandırma Doğruluğu: Doğruluk, sınıflandırma modellerinin performansını değerlendirmek için kullanılan bir metriktir ve modelin doğru yaptığı tahminlerin sayısını ifade eder. Basitçe şu şekilde ifade edilebilir:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Number\ of\ Total\ Predictions}$$

Ayrıca, basitçe şu şekilde de ifade edilebilir: $\frac{(TP+TN)}{(TP+TN+FP+FN)}$

Kesinlik Puanı: şu soruya cevap vermeye çalışır, pozitif tanımlamaların ne kadarı gerçekten doğruydu? TP, TN, FP, FN mantığını göz önünde bulundurarak şu şekilde ifade edebiliriz:

$$Precision = \frac{TP}{TP + FP}$$

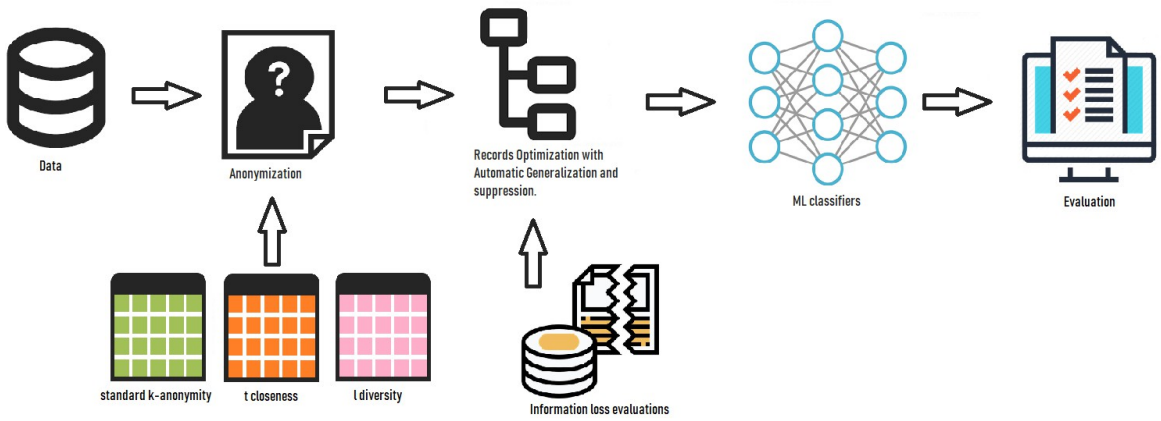
Hatırlama: Bu soruya cevap vermeye çalışır, gerçek pozitiflerin ne kadarı doğru tespit edildi? Ve şu şekilde ifade edilebilir:

$$Recall = \frac{TP}{TP + FN}$$

F1 Skoru: Hassasiyet ve Geri Çağırma değerleri arasında denge bulmak için kullanılan bir metriktir ve şu şekilde ifade edilebilir:

$$F1 = 2\,X\,\frac{precision * Recall}{precision + Recall}$$

4. ANONIMLEŞTIRME ÇERÇEVESI TASARIM



Şekil 4.1: Anonimleştirme çerçevesi

4.1 Genel Bakış

Şekil 4.1'de anonimleştirme çerçevesine ve değerlendirmesine genel bir bakış gösterilmekte olup, bu çerçeve aşağıdaki şekilde ayrıntılı olarak açıklanabilir.

- Verilere eriştikten sonra, yarı tanımlayıcılar ve hassas öznitelikler belirlenir. Bunlar seçilen *k* değeri ve kullanılacak *k-anonimlik* uzantısı ile birlikte ana dosyada saklanır.
- Anonimleştirilmiş kayıtlar Mondrian algoritması kullanılarak oluşturulur ve tüm genelleştirme ağaçları her sütun için tamamen otomatik bir şekilde oluşturulur.
- Optimizasyon adımı başlar ve her sütun için en iyi genelleştirme hiyerarşisini bulur ve sunulan bilgi kaybı metriğine göre en iyisi seçilir. Bu adımda gerekirse bastırma uygulanır. Otomasyon bu adımdan sonra sona erer.
- Elde edilen kayıtlar önceden işlenir ve Model Eğitimi için hazırlanır.
- Eğitim başlar ve değerlendirme metrikleri alınır.

Tüm adımları Python 3.7'de uyguladık. Bir çekirdek hizmetler dizinimiz ve otomatik genelleştirme ve bastırma hizmetleri dizinimiz var, burada ikinci dizin esnek ve

sadece benzer girdi ve çıktı yapısı sağlayarak herhangi bir yeni bastırma veya genelleştirme tekniğini kolayca entegre etmemizi sağlar. Ayrıca optimizasyon süreci sırasında çoklu işlem seçeneği de sunuyoruz, tüm bunlar kök dizindeki komut satırından kolayca çalıştırılabilen bir ana komut dosyası tarafından kontrol ediliyor.

4.2 Veri Setleri

Yetişkin veri kümesi: Bu veri kümesi 15 sütunlu 48843 veri noktası içermektedir. ABD 1994 nüfus sayımı veri tabanından elde edilmiştir. Yaş, iş sınıfı, eğitim, eğitim-num, maddi durum, meslek, ilişki, ırk, cinsiyet ve anavatan sütunlarını yarı tanımlayıcılar olarak ve gelir sütununu hedef sütun olarak kullandık.

Kaliforniya Konut Veri Kümesi: Bu veri kümesi 20640 veri noktası içermektedir. ABD 1990 nüfus sayımı verilerinden türetilmiştir. Enlem, boylam, konut medyan yaşı, medyan ev değeri ve medyan gelir sütunlarını yarı tanımlayıcılar olarak ve okyanus yakınlığını hedef sütun olarak kullandık.

Mamografik Kitle Veri Seti: Bu veri kümesi 6 sütunlu 830 veri noktası (temizlendikten sonra) içerir. Meme kanseri tarama sistemlerinden raporlanmıştır ve BI-RADS değerlendirmelerini içerir ve bir mamografik kitle lezyonunun ciddiyetini tahmin etmek için kullanılır. Tüm sütunları yarı tanımlayıcılar (yaş, şekil, BI-RADS değerlendirmesi, yoğunluk ve marj) ve hedef sütun olarak ciddiyet olarak kullandık.

5. MAKINE ÖĞRENIMI VE ANONIMLEŞTIRME SONUÇLARI

Önceki bölümlerde tartışılan ve Şekil 4.1'de özetlenen anonimleştirme tekniklerini uyguladık. Bu bölümde, her bir veri kümesi için elde edilen sonuçları ayrı ayrı tartışıyoruz. Orijinal veri kümelerine kıyasla k-anonimlik ve uzantılarını uyguladığımızda elde edilen veri kaybını açıklayarak başlıyoruz. Ardından, anonimleştirilmiş veri kümeleri kullanıldığında ML sonuçlarını yorumluyoruz.

5.1 Anonimleştirme ve bilgi kaybı Sonuçlar

Toplam bilgi kaybını, anonimleştirilmiş ve orijinal olmak üzere iki veri kümesi arasındaki farkı hesaplayarak hesaplıyoruz. Her iki veri kümesindeki hedef sütuna toplam sütun koşullu entropisini hesaplıyoruz ve anonimleştirilmiş veri kümesi sonuçlarının orijinal veri kümesine ne kadar benzer olduğunu ölçüyoruz.

Tablo 5.1: MGM veri kümesi: Hedef sütun 1Severity" ile entropi sonuçları (orijinal veri kümesi)

BI-RADS	Age	Shape	Margin	Density	Severity
0.361	0.221	0.266	0.279	0.005	1.0

Tablo 5.2: MGM veri kümesi: Hedef sütun "Önemlilik" ile entropi sonuçları (standart *k*-anonimlik)

BI-RADS	Age	Shape	Margin	Density	Severity
0.259	0.242	0.209	0.218	0.007	1.0

Örneğin, Tablo 5.1 orijinal MGM veri kümesinin her bir sütununun hedef sütunla birliktelik değerlerini (entropi katsayısı) ve Tablo 5.2 standart *k-anonimlik* uygulandıktan sonra MGM veri kümesi için birliktelik değerlerini göstermektedir. Ortalama değerleri karşılaştırdığımızda

ilişkilendirme sonuçlarına göre, anonimleştirme işleminin verilerin %82,6'sını korumayı başardığını ve bilgi kaybının %17,4 olduğunu gözlemliyoruz.

Genel olarak, beklenen teorik performans, standart versiyonun en iyi performansı göstereceğini ve l-çeşitliliğinin ikinci en iyi olması gerektiğini, t-yakınlığının ise en yüksek bilgi kaybına neden olacağını göstermelidir. Bunun nedeni, her bir uzantının *anonimleştirme* gruplarını bölme işlemine daha fazla kısıtlama eklemesi ve bunun sonucunda her bir grubun QI değerlerinin standart *k-anonimlikte* olacağından daha çeşitli olmasıdır. Her uzantının farklı gizlilik seviyesine sahip olduğunu ve *t-yakınlığının* en gizli olanı olduğunu unutmayın. Dolayısıyla, uzantı seçimi gizlilik ve veri kullanışlılığı arasında bir uzlaşmadır.

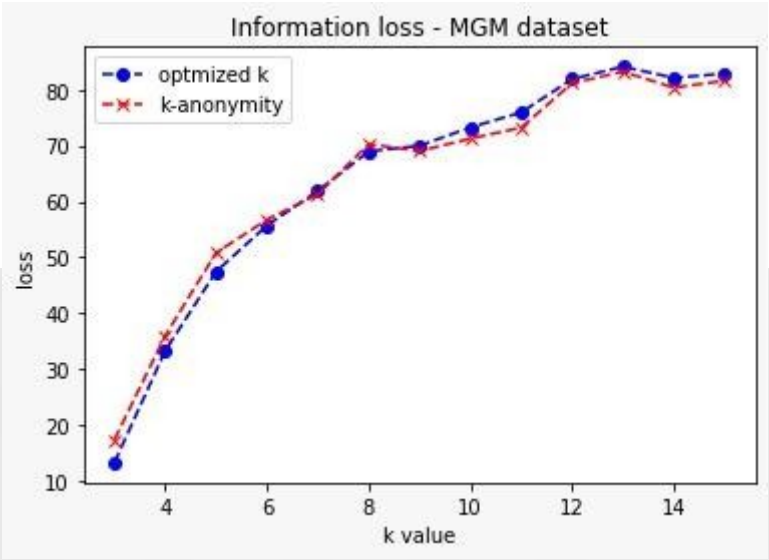
k-anonimlik ve uzantılarının davranışını daha iyi anlamak için, tüm veri kümelerine farklı koşullarda (örneğin, k=10, l=3, t=0.4) anonimleştirme uyguladık. Ardından, anonimleştirilmiş her veri kümesi için bilgi kaybını hesaplamaya devam ettik ve sonuçları Tablo 5.3, 5.4, 5.5 ve Şekil 5.1, 5.6, 5.11'de raporladık.

Şekil 5.1, 5.6 ve 5.11'de, standart ***k-anonimliği*** ve optimize edilmiş versiyonumuzu 3 ila 15 arasında değişen farklı k koşulları üzerinden tüm veri kümelerine uyguladık ve kayıp sonuçlarının yanı sıra bölüm sayısını da raporladık.

Tablo 5.3, 5.4, 5.5 standart *k-anonimlik,* optimize edilmiş versiyon, l-çeşitlilik ve *t-yakınlık* için sonuçları göstermektedir. Daha önce bahsedilen uzantıları her bir veri kümesine uyguluyoruz ve bilgi kaybı sonuçlarını ve bölüm sayısını raporluyoruz. Aşağıda her bir veri kümesi için bilgi kaybı sonuçlarını açıklıyoruz.

1) **MGM Veri Kümesi:**

MGM veri kümesi 830 girişe sahiptir ve elde edilen bölümleme grupları ve kayıp sonuçları Tablo 5.3 ve Şekil 5.1 - 5.5'te gösterilmektedir.

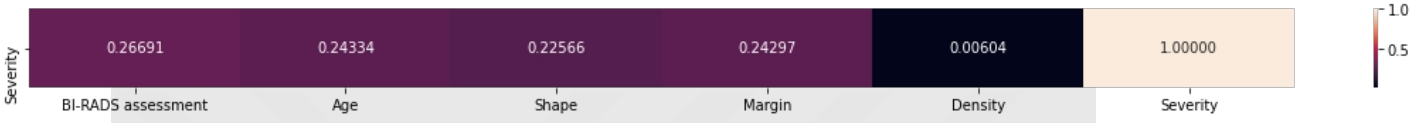


Şekil 5.1: MGM veri seti için farklı *k* değerleri üzerinden bilgi kaybı sonuçları

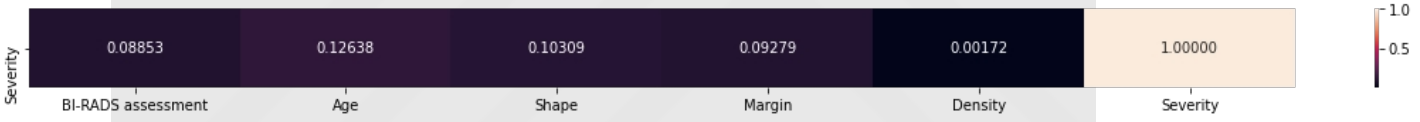
Tablo 5.3: MGM veri kümesi bölümleme ve bilgi kaybı sonuçları

MGM Data set		
	Partitions	Loss
Optimized k (k=3)	199	13.29%
k-anonymity (k=3)	199	17.38%
l-diversity (k=3, l=2)	125	63.50%
t-closeness (t=0,2)	33	98.37%
t-closeness (t=0,3)	74	89.71%
t-closeness (t=0,4)	119	65.37%

Şekil 5.2: MGM veri seti için entropi sonucu (anonimleştirme yok)



Şekil 5.3: MGM veri kümesi için entropi sonucu (standart ***k-anonimlik***)



Şekil 5.4: MGM veri seti için entropi sonucu (l-çeşitlilik)

Şekil 5.5: MGM veri seti için entropi sonucu (t-yakınlık)

Sonuçlar beklediğimiz teorik sonuçlarla tutarlıdır. En iyi performansın ***k-anonimlik,*** ikinci en iyi performansın *l-çeşitlilik ve* kötü performansın t-yakınlık olduğunu görüyoruz. *K* değerinin artmasının bölüm sayısını azalttığını ve bilgi kaybını artırdığını görüyoruz. *l-çeşitliliği* de daha kısıtlayıcı doğası nedeniyle büyük bilgi kaybı gösteriyor. *t-yakınlık* sonuçları *t* değerini değiştirdiğimizde değişiyor.

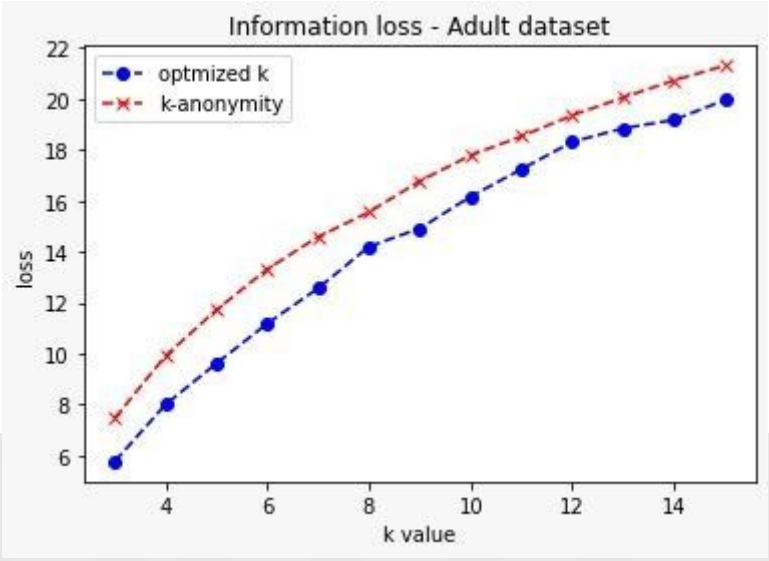
T-yakınlığı için varsayılan uzaklık değerimiz 0,2'dir ve bu değer genişletme kriterlerini garanti etmek için çok katıdır. *t-yakınlığının,* gruplandırılmış her veri kaydında orijinal veri kümesinin dağılımıyla aynı veri dağılımına sahip olmaya çalışmak için bir mesafe ölçümü kullandığını biliyoruz. Ayrıca daha yüksek *t* değerleri kullandık ve *t* değerini artırdıkça daha az gizlilik ve daha az bilgi kaybına yol açan daha az kısıtlama olacaktır. Optimize edilmiş ***k-anonimliğimizden*** elde edilen sonuçlar, standart ***k-anonimlik ile*** karşılaştırıldığında bilgi kaybını biraz azaltmış gibi görünmektedir.

2) Yetişkin Veri Kümesi

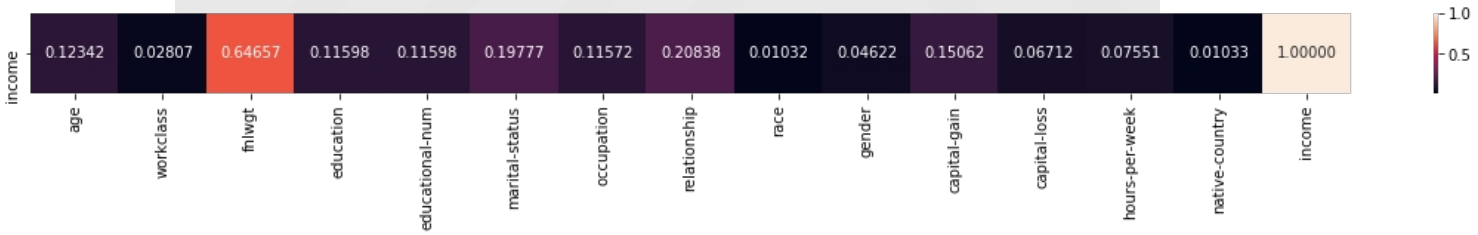
Yetişkin veri kümesinde 48843 giriş vardır ve elde edilen bölümleme grupları ve kayıp sonuçları Tablo 5.4 v e Şekil 5.6 - 5.10'da gösterilmiştir.

Tablo 5.4: Yetişkin veri kümesi bölümleme ve bilgi kaybı sonuçları

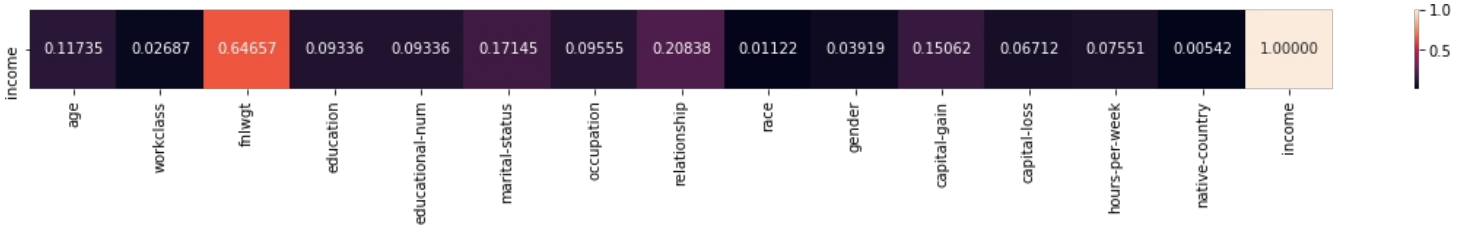
Adult Data set		
	Partitions	Loss
Optimized k (k=3)	9246	5.76%
k-anonymity (k=3)	9246	7.46%
l-diversity (k=3, l=2)	4919	9.29%
t-closeness (t=0,2)	2925	32.66%
t-closeness (t=0,3)	7134	7.92%
t-closeness (t=0,4)	7863	8.07%



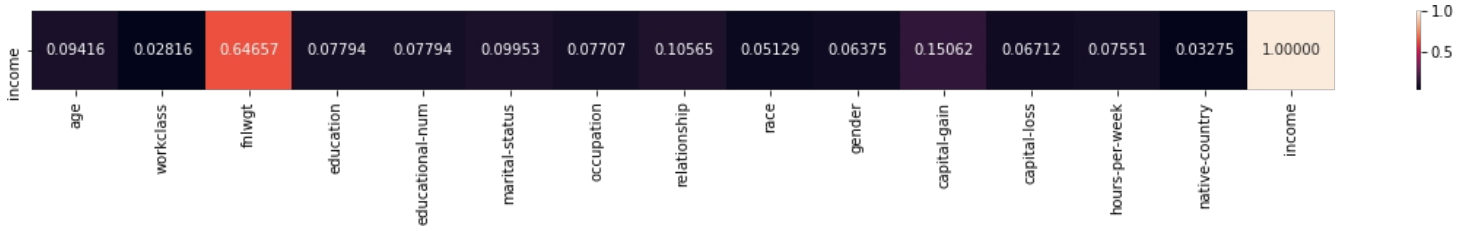
Şekil 5.6: Yetişkin veri seti için farklı *k* değerleri üzerinden bilgi kaybı sonuçları



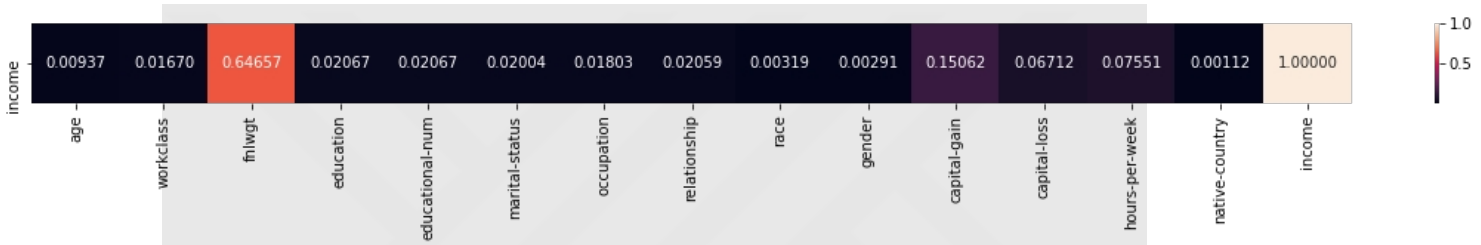
Şekil 5.7: Yetişkin veri seti için entropi sonucu (anonimleştirme yok)



Şekil 5.8: Yetişkin veri kümesi için entropi sonucu (standart *k-anonimlik*)



Şekil 5.9: Yetişkin veri seti için entropi sonucu (l-çeşitlilik)



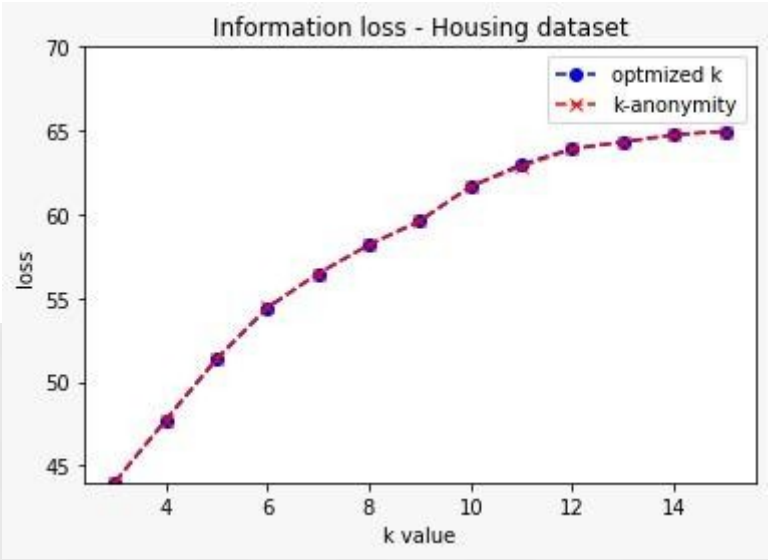
Şekil 5.10: Yetişkin veri seti için entropi sonucu (t-yakınlık)

Yetişkin veri kümesi, *k-anonimlik* çalışmalarında en sık kullanılan veri kümelerinden biridir. Tablo 5.4'te gösterilen sonuçlardan teorik beklentilerimizin tüm uzantılarda karşılandığını görebiliriz. *K* değeri ne kadar yüksek olursa bilgi kaybının da o kadar fazla olduğunu görüyoruz. Benzer şekilde, *l-çeşitlilik* uzantısını benimsediğimizde bilgi kaybı artmakta ve en yüksek kayıp *t-yakınlık* uygulandığında gözlenmektedir. Bununla birlikte, bilgi kaybı diğer veri kümelerine kıyasla minimum düzeydedir ve bu veri kümesinin anonimleştirme için çok uygun olduğu görülmektedir. Anonimleştirme işlemi sırasında çok az genelleme ve bastırma gerektirmektedir. Bu sonuçlar, muhtemelen karşılaştığımız çalışmaların çoğunun neden *k-anonimlik* analizi için bu veri kümesini kullandığını da açıklamaktadır. Optimize edilmiş *k-anonimliğimizden* elde edilen sonuçlar, standart *k-anonimlik* ile karşılaştırdığımızda bilgi kaybında gözle görülür bir azalma olduğunu göstermektedir.

3) Konut Veri Seti

Konut veri kümesinde 20640 giriş vardır ve her bir varyasyon için ortaya çıkan bölüm

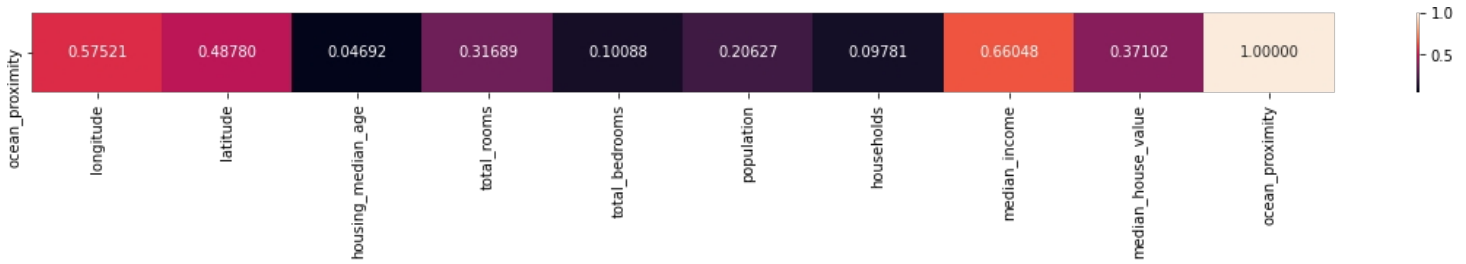
grupları ve kayıp değerleri Tablo 5.5 ve Şekil 5.11 - 5.15'te gösterilmektedir.



Şekil 5.11: Konut veri seti için farklı k değerleri üzerinden bilgi kaybı sonuçları

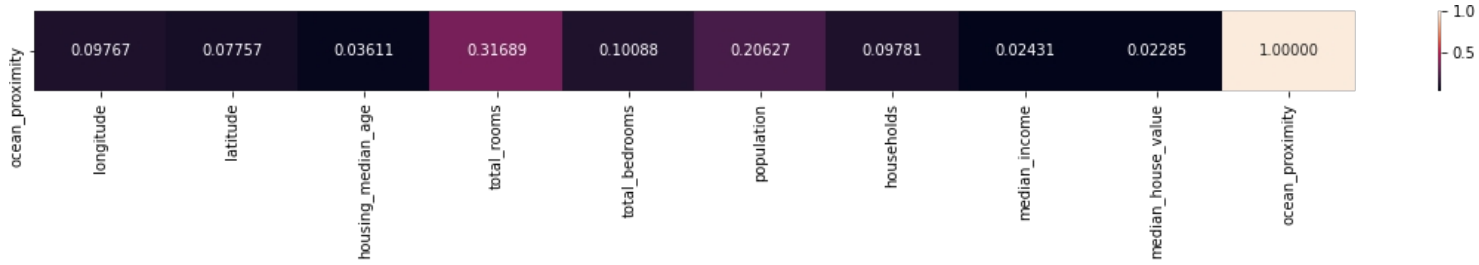
Tablo 5.5: Konut veri kümesi bölümleme ve bilgi kaybı sonuçları

Housing Data set		
	Partitions	Loss
Optimized k (k=3)	5351	43.99%
k-anonymity (k=3)	5351	44.02%
l-diversity (k=3, l=2)	2391	65.83%
l-diversity (k=3, l=3)	769	71.82%
t-closeness (t=0,2)	131	67.42%
t-closeness (t=0,3)	538	66.52%
t-closeness (t=0,4)	1040	65.32%

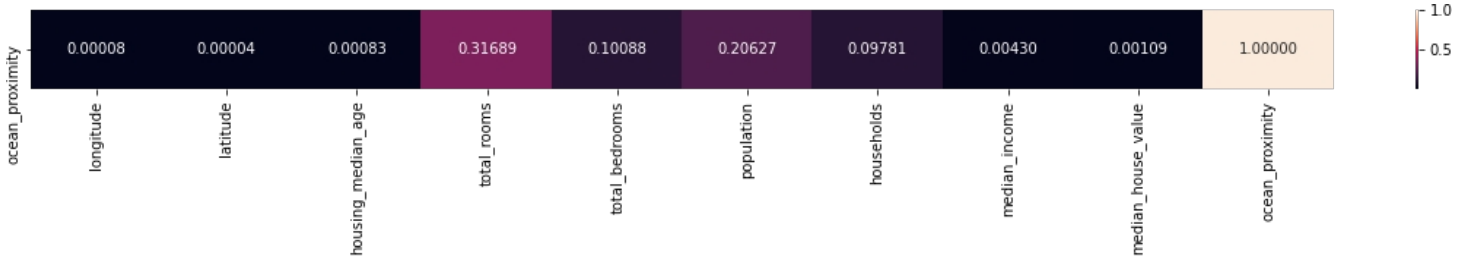


Şekil 5.12: Konut veri seti için entropi sonucu (anonimleştirme yok)

Şekil 5.13: Konut veri seti için entropi sonucu (standart ***k-anonimlik***)



Şekil 5.14: Konut veri seti için entropi sonucu (l-çeşitlilik)



Şekil 5.15: Konut veri seti için entropi sonucu (t-yakınlık)

Konut veri kümesi, 1H-OKYANUS için 0,44, INLAND için 0,317 ve NEAR-OCEAN için 0,23 frekanslarıyla 3 benzersiz hedef sütun değerine sahiptir. Tüm uzantılarda önemli bir veri kaybı olduğunu görüyoruz. Bunun başlıca nedeni ilk iki QI olan boylam ve enlemin genelleştirme sürecidir. Bu davranış diğer makalelerde de rapor edilmiştir [19]. Bazı veri kümelerinde, anonimleştirme daha yüksek bilgi kaybına neden olabilir. Daha açık olmak gerekirse, bir veri kümesindeki bazı veri sütunu değerleri diğerlerine göre önemli ölçüde bağımlıdır ve hedef sütun değerleriyle yüksek korelasyona sahiptir, bu da genelleştirmeyi ve aynı zamanda veri gizliliğini korumayı çok zorlaştırır. Farklı k, l ve t değerlerini denediğimizde benzer bir davranış fark ederken, bu veri kümesinin anonimleştirme işleminden sonra en yüksek bilgi kaybına maruz kaldığı sonucuna vardık.

5.2 Makine Öğrenimi Sınıflandırıcıları Sonuçlar

Anonimleştirmenin etkisini anlayabilmek için öncelikle modellerimizi anonimleştirme olmadan orijinal veri kümeleri üzerinde eğitmeye, performanslarını raporlamaya ve her veri kümesi için anonimleştirilmiş verilerle eğitilmiş modellerle karşılaştırmak için kullanmaya karar verdik.

Tablo 5.6: Orijinal veri kümeleri makine öğrenimi sonuçları

Model	MGM		Housing		Adult	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
Extra Trees	0.767	0.766	**0.963**	**0.963**	**0.858**	**0.788**
Random Forest	0.779	0.778	**0.974**	**0.974**	**0.857**	**0.791**
Gradient Boosting	0.771	0.771	**0.985**	**0.985**	**0.874**	**0.816**
SVM	0.779	0.778	0.612	0.612	0.760	0.432
Logistic Regression	**0.819**	**0.819**	0.596	0.532	0.789	0.601
SGD	0.526	0.481	0.594	0.450	0.239	0.193
GaussianNB	0.790	0.790	0.664	0.635	0.794	0.645
KNN	**0.823**	**0.822**	0.861	0.854	0.854	0.789
ANN	**0.791**	**0.790**	0.588	0.540	0.742	0.591

Tablo 5.7: Veri kümeleri makine öğrenimi sonuçları, optimizasyonlu *k-anonimlik*

Model	MGM		Housing		Adult	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
Extra Trees	0.726	0.722	**0.762**	**0.751**	**0.830**	**0.754**
Random Forest	**0.738**	**0.737**	**0.772**	**0.761**	**0.840**	**0.767**
Gradient Boosting	**0.767**	**0.765**	**0.783**	**0.771**	**0.863**	**0.795**
SVM	0.718	0.716	0.476	0.305	0.760	0.432
Logistic Regression	0.738	0.737	0.490	0.399	0.799	0.644
SGD	0.670	0.643	0.458	0.242	0.761	0.435
GaussianNB	**0.742**	**0.742**	0.501	0.451	0.793	0.643
KNN	0.730	0.729	0.558	0.499	**0.838**	**0.766**
ANN	**0.767**	**0.766**	0.457	0.413	0.738	0.583

Tablo 5.8: Yetişkin veri kümeleri makine öğrenimi sonuçları, optimizasyon olmadan

Model	k-anonymity ($k = 3$)		l-diversity ($l = 2$)		t-closeness ($t = 0, 2$)	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
Extra Trees	**0.830**	**0.753**	**0.797**	**0.703**	**0.784**	**0.647**
Random Forest	**0.837**	**0.764**	**0.807**	**0.715**	**0.775**	**0.672**
Gradient Boosting	**0.865**	**0.800**	**0.848**	**0.757**	**0.831**	**0.707**
SVM	0.760	0.432	0.760	0.432	0.760	0.432
Logistic Regression	0.788	0.595	0.794	0.609	0.799	0.641
SGD	0.760	0.432	0.761	0.433	0.761	0.434
GaussianNB	0.790	0.640	0.794	0.643	0.792	0.644
KNN	**0.840**	**0.767**	**0.820**	**0.722**	**0.800**	**0.664**
ANN	0.729	0.573	0.735	0.578	0.741	0.575

Tablo 5.9: Mamografik kitle veri kümeleri makine öğrenimi sonuçları, optimizasyon olmadan

Model	**k-anonymity** ($k = 3$)		l-diversity ($l = 2$)		t-closeness ($t = 0, 2$)	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
Extra Trees	0.746	0.746	0.654	0.651	0.546	0.511
Random Forest	**0.767**	**0.766**	**0.686**	**0.685**	**0.558**	**0.471**
Gradient Boosting	**0.767**	**0.766**	0.666	0.664	0.546	0.511
SVM	0.714	0.713	**0.730**	**0.730**	**0.558**	**0.471**
Logistic Regression	0.694	0.693	0.642	0.635	0.558	0.471
SGD	0.526	0.481	0.570	0.566	0.550	0.480
GaussianNB	0.694	0.693	0.646	0.640	0.534	0.515
KNN	0.694	0.693	0.646	0.641	**0.558**	**0.471**
ANN	**0.759**	**0.759**	**0.714**	**0.713**	0.502	0.449

Tablo 5.10: Kaliforniya Konut veri kümeleri makine öğrenimi sonuçları, optimizasyon olmadan

Model	k-anonymity ($k = 3$)		l-diversity ($l = 2$)		t-closeness ($t = 0, 2$)	
	Accuracy	F1	Accuracy	F1	Accuracy	F1
Extra Trees	**0.765**	**0.755**	**0.625**	**0.599**	**0.470**	**0.419**
Random Forest	**0.770**	**0.759**	**0.624**	**0.597**	**0.467**	**0.424**
Gradient Boosting	**0.783**	**0.772**	**0.652**	**0.626**	**0.481**	**0.416**
SVM	0.475	0.303	0.465	0.273	0.462	0.271
Logistic Regression	0.499	0.404	0.558	0.483	0.470	0.373
SGD	0.451	0.229	0.346	0.309	0.452	0.248
GaussianNB	0.499	0.459	0.567	0.546	0.361	0.284
KNN	0.558	0.506	0.617	0.577	0.468	0.376
ANN	0.458	0.411	0.451	0.404	0.442	0.394

5.3 Modellerine İlişkin Genel Açıklamalar

Makine Öğrenimi modellerinin sonuçları Tablo 5.3 - 5.7'de gösterilmiştir ve veri kaybı sonuçlarıyla tutarlıdır. Genel olarak, ağaçlar ve topluluk tabanlı modeller anonimleştirilmiş veri kümelerinde daha iyi performans gösterme eğilimindedir ve en iyi performans gösteren modeller esas olarak Gradient Boosting, Random Forests ve Extra Trees'tir. Bu üç modelin davranışının tutarlı kaldığını ve tüm veri kümelerinde en iyi performansı gösterdiklerini görüyoruz. Bunun nedeni, ağaç tabanlı modellerin eksik veya normalize edilmemiş verilerle iyi çalışma eğiliminde olmaları ve farklı değişkenler arasındaki bağlantıları tespit etmede iyi olmalarıdır. Bu nedenle, genelleştirilmiş ve bastırılmış verilerde iyi performans göstermişlerdir. Ayrıca YSA'nın MGM veri setinde tüm uzantılarda iyi performans gösterdiğini, diğer veri setlerinde ise kötü performans gösterdiğini gözlemliyoruz. Bu davranışın parametre ayarlamasıyla ilgili olduğunu düşünüyoruz. Hem SVM hem de KNN modelleri Yetişkin ve Konut veri kümelerinde kötü performans göstermiştir ve bunun temel nedeni genelleştirilmiş verilerle başa çıkma şekilleridir, ancak MGM veri kümesinde tüm uzantılarda tutarlı bir şekilde daha iyi performans göstermişlerdir.

Stokastik gradyan inişi (SGD) ve lojistik regresyon sınıflandırıcıları Doğrusal modeller ailesine aittir ve Yetişkin ve Konut veri kümelerinde kötü performans göstermişlerdir.

Lojistik Regresyon sınıflandırıcısının MGM'de diğer modellere biraz yakın performans gösterdiğini görüyoruz

standart *k-anonimlikte* veri kümesi. SGD'nin davranışı, orijinal veri kümelerinde de kötü performans gösterdiği için bekleniyordu. Bu modellerin doğrusal yapıları ve genelleştirilmiş verilerle başa çıkamamaları nedeniyle başarısız olduklarına inanıyoruz. Gaussian Naive Bayes de Konut ve Yetişkin veri kümelerinde kötü, MGM veri kümesinde ise tutarsız performans göstermiştir. Anonimleştirilmiş ve genelleştirilmiş veri kümeleriyle uğraşırken en iyisi olmadığını fark ettik.

Bilgi kaybı metriğinin ML'de anonimleştirilmiş veri performansına ilişkin içgörü sağlamada çok yararlı olmasına rağmen, gerçek ML sonuçlarıyla karıştırılmaması gerektiğini belirtmek önemlidir. Sınıflandırıcıların sonuçlarının açıklanabilir olduğunu ve kayıp sonuçlarıyla güçlü bir şekilde bağlantılı olduğunu görüyoruz. Örnek olarak, Yetişkinler veri kümesi için kayıp sonuçlarına ve sınıflandırıcı sonuçlarına baktığımızda, bunların güçlü bir şekilde bağlantılı olduğunu gözlemliyoruz. Başka bir örnek olarak, MGM veri kümesinde, t = 0.2 ile *t-yakınlık* uzantısı kullanıldığında, kayıp %98'e ulaşır, bu da neredeyse tüm ilişkinin kaybolduğu anlamına gelir. Bununla birlikte, sınıflandırıcıların doğruluk puanları %50 civarındadır. Bu beklenmedik bir sonuç değildir, çünkü MGM veri kümesi hedef sütununda her biri yaklaşık %52 ve %48 değer sıklığına sahip iki benzersiz değere sahiptir. Sınıflandırıcılar bir sınıfı diğerine göre tahmin etmeye devam edebilir ve bu da rapor edilen sonuçlara neden olur. Genel olarak, elde edilen test sonuçları, kullanılan kayıp metriğinin bir veri kümesinin ML performansı için doğru bir gösterge olduğunu ve anonimleştirilmiş verilerin ML uygulamaları için uygunluğunun önceden bir göstergesi olarak kullanılabileceğini göstermektedir.

Bazı durumlarda, bilgi kaybındaki azalmanın önemli olmaması nedeniyle optimize edilmiş ve normal k-anonimlik için benzer doğruluk sonuçları görüyoruz. Bununla birlikte, daha iyi modellerle *k-anonimliğin* optimizasyonla ayarlanmasının, bilgi kaybındaki farkı yansıtan umut verici sonuçlar gösterebileceğine inanıyoruz.

Bu çalışmada, verilerin genelleştirilmesi ve bastırılması en kritik kısımlar olmak üzere, anonimleştirilmiş verilerin kullanılabilirliği üzerinde önemle durulmaktadır. Verileri minimum kayıpla daha iyi genelleştirmek için, bir bilgi kaybı metriği ile entegre edilmiş

otomatik bir genelleştirme yaklaşımı önerilmiştir.

Sonuçlarımızı diğer çalışmalarla karşılaştırdığımızda, örneğin [19]'da Yetişkin veri kümesinde onlardan belirgin bir şekilde daha iyi performans gösterdiğimizi, MGM veri kümesinde biraz daha yüksek sonuçlar elde ettiğimizi ve Konut veri kümesinde genel olarak benzer sonuçlar elde ettiğimizi görüyoruz. 18]'de sunulan Mondrian sonuçlarıyla karşılaştırdığımızda, yine yaklaşımımızın Yetişkin veri kümesinde daha iyi performans gösterdiğini ve MGM veri kümesinde çok benzer performans gösterdiğini gözlemliyoruz. Öte yandan, önerdikleri NSVDist yaklaşımıyla karşılaştırıldığında, yaklaşımımız Yetişkin veri kümesinde biraz daha iyi performans gösterirken MGM veri kümesinde yine oldukça benzer bir performans sergiliyor. Sonuçlarımız, önerilen otomatik anonimleştirme yaklaşımının önceki standart yaklaşımlardan daha iyi performans gösterebileceğini veya bazı durumlarda onlarla aynı seviyede olabileceğini göstererek umut verici görünmektedir.

5.4 k-anonimlik üzerine genel açıklamalar varyasyonlar

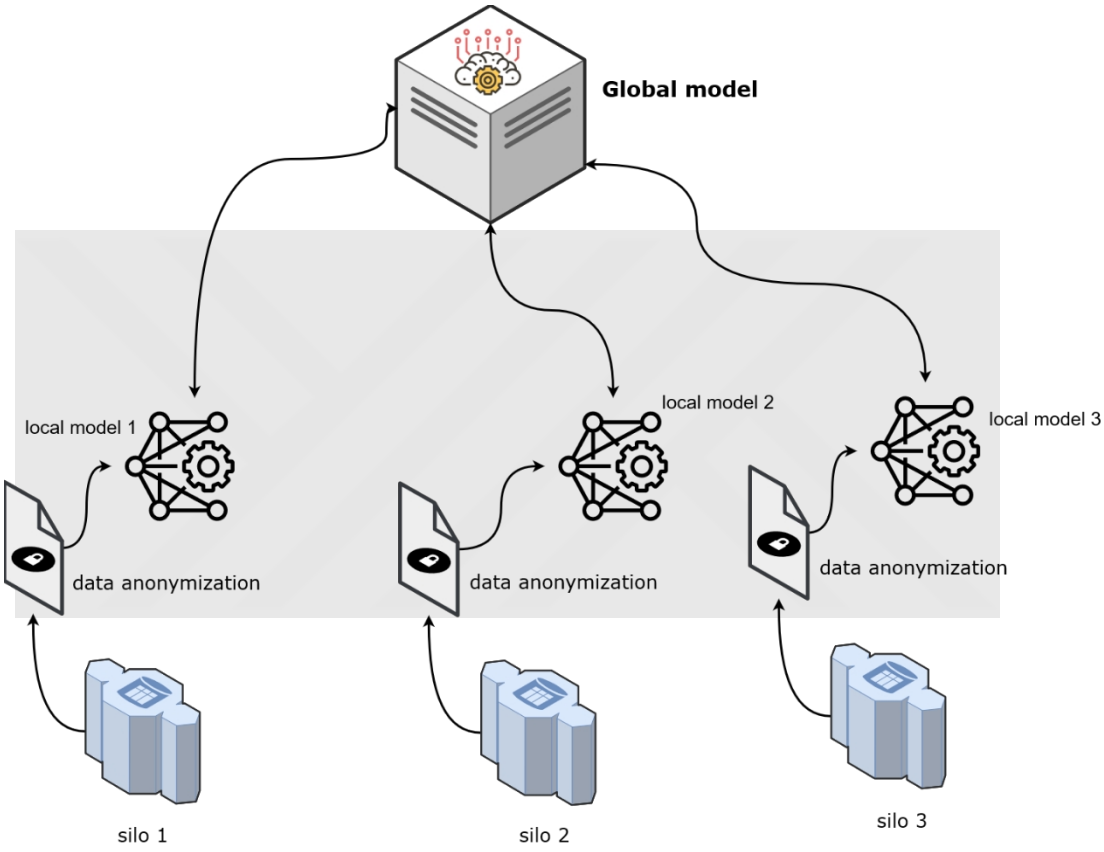
Daha önce, her bir uzantı için teorik olarak beklenen sonuçlara sahip olduğumuzdan bahsetmiştik; anonimleştirme işlemi ne kadar çok kısıtlamaya tabi tutulursa o kadar çok veri kaybı yaşanır. *l-çeşitlilik kısıtlamaları,* her bir gruplanmış veri kaydındaki hedef sütun değerlerinin sayısını belirleyen *l değerinin* seçiminden kaynaklanır ve bu değer anonimleştirme işlemine ekstra kısıtlamalar getirerek performansı etkiler. t-yakınlık kısıtlamaları, orijinal veri kümesi hedef değer dağılımı ile her bir kayıt dağılımı arasında seçilen kabul edilen uzaklık kriterinden kaynaklanır. Her uzantı için çeşitli koşul değerlerini denedik ve kayıp sonuçlarını gösterdik ve teorik olarak beklenen sonuçlarımızın karşılandığı sonucuna vardık.

Bu çalışmada, ML modellerini eğitmek için sonuç verilerini kullanmamız gerektiğinde, her bir uzantıdaki koşullar için varsayılan değerleri kullandık. Varsayılan koşul değerlerimiz *k* = 3, *l-çeşitlilik için l* = 2 ve *t-yakınlık* uzantısı için 0,2 uzaklık ölçütüdür. Seçtiğimiz *l değeri*, l-çeşitlilik özelliğini sağlayan en az kısıtlayıcı değer olma eğilimindedir ve aynı zamanda benzersiz hedef sınıf değerlerinin sayısından daha yüksek olmaması gerektiğine dikkat etmek önemlidir. *t-closeness* uzantısında mesafe ölçümü için 0.2 seçimi, gruplandırılmış veri kayıtlarını sınırlar ve maksimum gizlilik elde etmek için ekstra bir kısıtlama katmanı ekler ve her kaydın dağılımının orijinal veri kümesine çok benzer

olmasını sağlar, değer ne kadar büyükse dağılımlar arasındaki mesafe o kadar büyük olur. Uzaklık değerinin seçimi, dağılımın

bölme işlemi, anonimleştirme işlemi, bilgi kaybı ve gizlilik seviyesi ve değeri dikkatlice seçilmelidir.

6. ANONİMLEŞTİRME ÇERÇEVESİ İLE SİLOLAR ARASI FEDERASYONLU ÖĞRENME



Şekil 6.1: Anonimleştirme çerçevesi iş akışıyla çapraz Silo federasyonlu öğrenme.

Bu çalışmada, Şekil 6.1'de gösterildiği gibi, gizliliği artırmak ve olası saldırılara karşı korunmaya yardımcı olmak için federe öğrenme için kullanılacak veriler üzerinde k-anonimliği araştırıyor ve uyguluyoruz. Çapraz silo federasyonlu öğrenme, eğitim sürecini sürdürmek için az sayıda güvenilir istemciye dayanır. Çoğunlukla eğitim amacıyla güvenli ve kontrollü ortamlara sahip büyük kuruluşlarda ve hastanelerde kullanılır. Şimdi çerçeve bileşenlerini tartışacağız.

6.1 Veri Dağıtım

Federe bir öğrenme iş akışını simüle etmek için verileri farklı istemciler arasında dağıtmamız gerekir. Bu çalışmada UC Irvine Machine Learning Repository'den Census Income veri setini kullandık. Veri kümesi 48843 kayıt ve 15 sütun içermektedir.

Veri dağıtım stratejimiz aşağıdaki gibidir:

- Tüm veri setini, test setinin verilerin %30'unu içerdiği ve geri kalanının eğitim için tutulduğu test ve eğitim setleri olmak üzere iki sete bölerek başlıyoruz.
- Eğitim setini alıyoruz ve önceden kararlaştırılan sayıda bölüme ayırıyoruz, bölüm sayısının kullanmayı hedeflediğimiz istemci sayısıyla aynı olduğuna dikkat ediyoruz.
- Eğitim bölümleri anonimleştirme sürecine yönlendirilir.

6.2 *k*- anonimlik

Mondrian tabanlı bir ***k-anonimlik*** yaklaştırma algoritması kullanıyoruz [46]. *K* kriterinin karşılandığını garanti etmek için hem genelleme hem de bastırma kullanıyoruz. Mondrian'ın ***k-anonimleştirme*** için en iyi performans gösteren algoritma olduğu bildirilmektedir [47]. Her sütun için genelleştirme ağaçları oluşturan ve her bölüm için veri kaybını en aza indiren ve ***k-anonimliği sağlayan en*** iyi genelleştirme ağacını bulmak için bir bilgi kaybı metriği kullanan otomatik bir genelleştirme yaklaşımı geliştirdik. Önemli bir adım, anonimleştirme işlemi için yarı tanımlayıcıların seçilmesidir. Veri setlerinden, bir araya getirildiklerinde deneklerin doğrudan tanımlanmasını sağlayabileceğine inandığımız bir özellik alt kümesi seçiyoruz. Elimizdeki her eğitim bölümü için önceki yaklaşımı uyguluyoruz ve istemciler arasında veri dağıtımına geçiyoruz.

6.3 Federated Learning İstemciler

Her eğitim bölümü için anonimleştirme işlemini tamamladıktan sonra, her bölüm için bir istemci oluşturuyoruz. Her bir istemci bölümü işlenecek ve eğitim için hazırlanacaktır. Her bölüm, 32'lik bir parti boyutuna sahip bir tfds nesnesine (TensorFlow veri oluşturucu) dönüştürülecektir.

6.4 Federe Öğrenme Küresel ve Yerel Modelleri

Hem global hem de yerel modeller aynı model mimarisini paylaşmaktadır. Çok Katmanlı Algılama (MLP) algoritması kullanıyoruz. Bunlardan 3'ü gizli katman, diğer ikisi sırasıyla giriş ve çıkış katmanları olmak üzere 5 katmanlı bir mimari kullandık. Optimizasyon algoritması olarak 0.01 öğrenme oranı ve Stokastik gradyan inişi (SGD) [48] kullandık.

6.5 Modeller Toplama

49]'dan türetilen yinelemeli bir model ortalama algoritması kullanarak yerel modellerin ağırlıklarını küresel modellere güncelliyoruz. Bahsedilen yaklaşım bu denklem ile elde edilir:

$$f(w) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(w).$$

Nerede?

$$_k F(w) = \frac{1}{n_k} \sum_{i \in pk} f_i(w).$$

Bölümlenmiş kümelerimizi hazırladıktan ve istemciler arasında dağıttıktan sonra. Eğitim sürecini şu şekilde başlatıyoruz:

- Küresel bir model oluşturarak başlıyoruz, ardından yerel modeller oluşturmaya başlıyoruz ve bunları her müşteriye atıyoruz.

- Eğitim sürecini başlatıyoruz, her epokta (eğitim turu) yerel modeller ağırlıklarını güncellenmesi için global modele gönderiyor.

- Global modeli güncellemeden önce her epokta (turda) yerel modellerin ağırlıkları için federe öğrenme ortalaması uyguluyoruz.

- Küresel model her epokta (turda) güncellendikten sonra, küresel model ağırlıklarını yerel modellerle paylaşır.

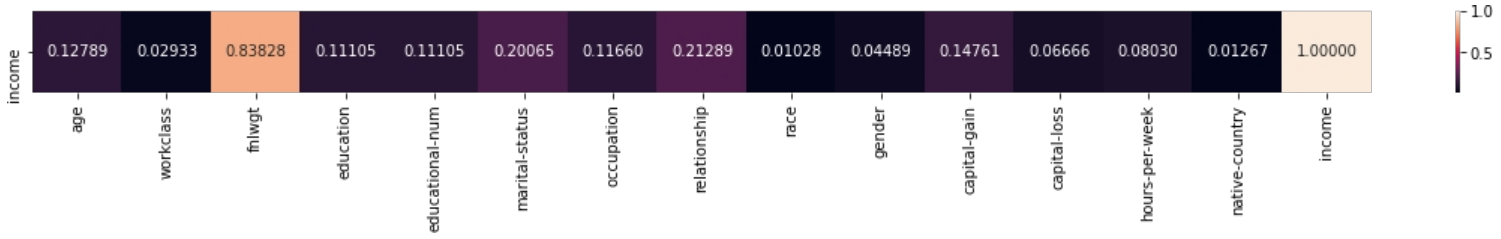
- Her bir global model ile anonimleştirilmemiş test bölümü üzerinde test yaparak global doğruluğu hesaplıyoruz.

7. FEDERASYON ÖĞRENME ÇERÇEVESİ SONUÇLAR

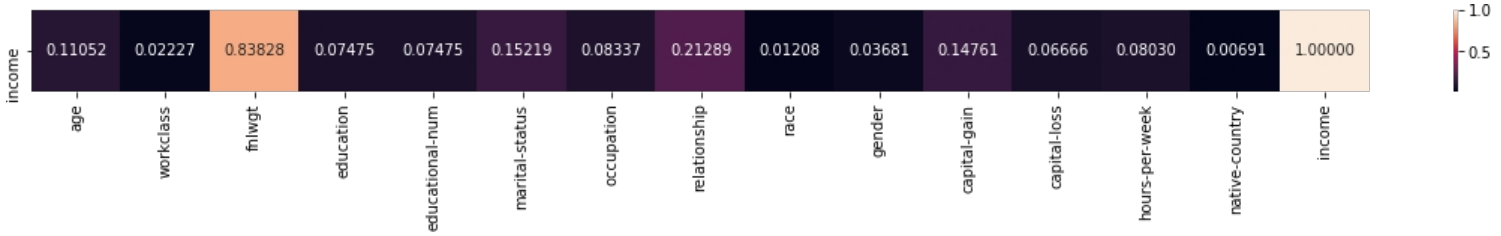
Sonuçları karşılaştırabilmek için, daha önce bahsedilen yaklaşımı aynı eğitim süreci ve müşterinin veri dağıtım koşulları altında anonimleştirilmemiş veriler ve anonimleştirilmiş veriler üzerinde uygulamamız gerekir. Ayrıca eğitimin sonunda global doğruluk, kesinlik, hatırlama ve F1 puanını hesaplayacağız. Küresel doğruluk, küresel modelin tüm istemcilerde tüm eğitim dönemlerini (turlarını) tamamladıktan ve güncellemeyi bitirdikten sonra son sürümü üzerinde yapılan testlerden elde edilen doğruluğu ifade eder. Verileri 3 silo arasında bölüyoruz, veri kümesi 3 parçaya bölünüyor, ardından anonimleştirme aşamasına taşınıyor. Test kümemiz olarak anonimleştirilmemiş bir bölüm kullanıyoruz. Her bir veri bölümünü grup1, grup2 ve grup3 olarak adlandıracağız.

7.1 Bilgi Kaybı Sonuçlar

Grup1, grup2 ve grup3 için her bir sütuna ilişkin bilgi kaybı sonuçları Şekil 7.1 - 7.6'da gösterilmektedir.

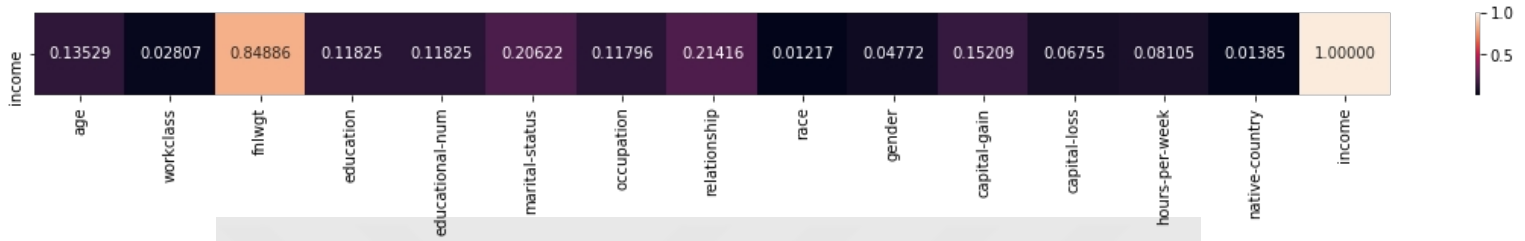


Şekil 7.1: Anonimleştirmeden önce Grup 1 için bilgi kaybı sonuçları



Şekil 7.2: Anonimleştirme sonrası Grup 1 için bilgi kaybı sonuçları

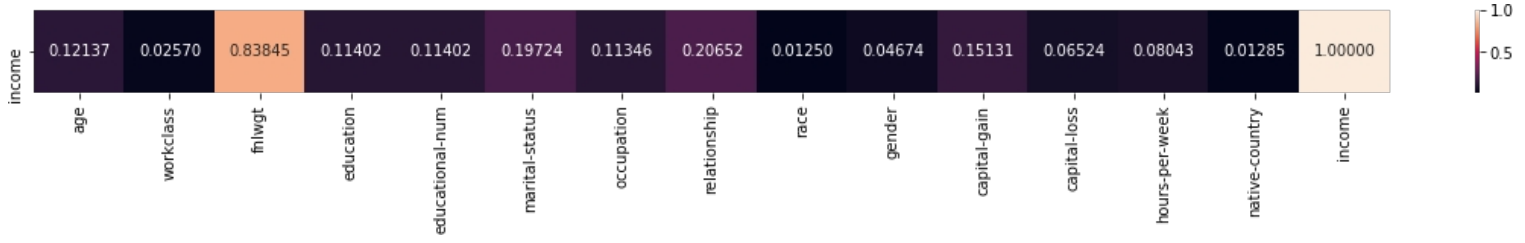
Her bir sütun için bilgi kaybı sonucunu karşılaştırırsak (her birini orijinal değeriyle karşılaştırarak) %6,13'lük bir bilgi kaybı fark ederiz, dolayısıyla bilginin %93,87'sini koruyabilmiştir.



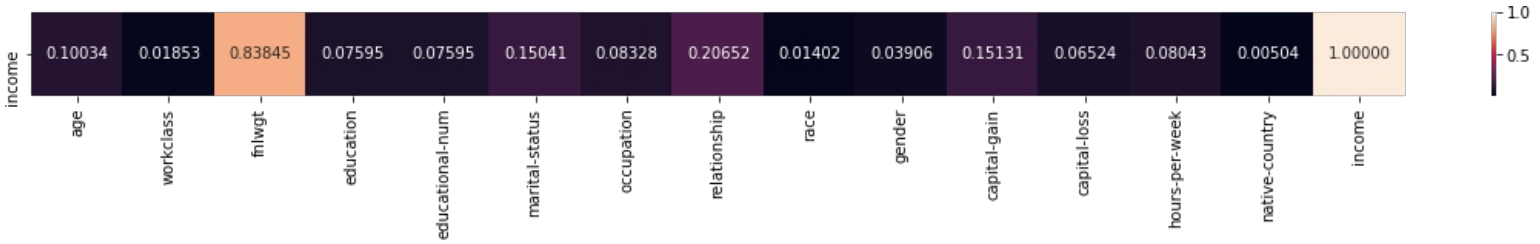
Şekil 7.3: Grup 2 için anonimleştirme öncesi bilgi kaybı sonuçları

Şekil 7.4: Anonimleştirme sonrası Grup 2 için bilgi kaybı sonuçları

Her bir sütun için bilgi kaybı sonucunu karşılaştırırsak (her birini orijinal değeriyle karşılaştırarak) %5,97'lik bir bilgi kaybı olduğunu fark ederiz, dolayısıyla bilginin 94,03'ünü koruyabilmiştir.



Şekil 7.5: Grup 3 için anonimleştirme öncesi bilgi kaybı sonuçları



Şekil 7.6: Anonimleştirme sonrası Grup 3 için bilgi kaybı sonuçları

Her bir sütun için bilgi kaybı sonucunu karşılaştırırsak (her birini orijinal değeriyle karşılaştırarak), bilginin %93,70'ini koruyabilirken %6,30'luk bir bilgi kaybı olduğunu fark ederiz.

7.2 Eğitim Sonuçlar

Tablo 7.1 anonimleştirilmemiş veriler için FL sonuçlarını gösterirken, Tablo 7.2 anonimleştirilmiş veriler için FL sonuçlarını göstermektedir.

Tablo 7.1: Anonimleştirilmemiş veri federasyonlu öğrenme sonuçları

Global Accuracy	Precision	Recall	f1 score
0.850	0.727	0.616	0.667

Tablo 7.2: Anonimleştirilmiş veri federasyonlu öğrenme sonuçları

Global Accuracy	Precision	Recall	f1 score
0.835	0.695	0.561	0.621

7.3 Federe öğrenme çerçevesi açıklamalar

Anonimleştirilmiş veri FL modelinin küresel doğrulukta hafif bir düşüşe sahip olduğunu fark edebiliriz. Bu durum F1 skoruna ve hesapladığımız diğer metriklere de yansımaktadır. Bu düşüş anonimleştirme işleminin bir sonucudur. Anonimleştirme işlemi bir dereceye kadar bilgi kaybına yol açtığı için bu davranış beklenen bir durumdur. Uygun genelleme ve bastırma teknikleriyle anonimleştirmenin veri kaybını en aza indirebileceği sonucuna varabiliriz. FL'ye anonimleştirmenin eklenmesi, bunlardan herhangi birinin tek başına kullanılmasından daha fazla gizlilik sağlar.

Bu makalede ele aldığımız saldırılar, homojenlik saldırıları ve arka plan bilgisi saldırıları, bir saldırganın anonimleştirilmiş verilere erişebileceği ve belirli konular hakkında bilgi edinmek için k-anonimlik üzerindeki bazı sınırlamaları istismar ederek veri gizliliğini tehlikeye atabileceği varsayımı altında çalışır. Bu tür saldırılar, ham veya anonimleştirilmiş veri paylaşılmadığından, bunun yerine yerel modellerin ağırlıkları paylaşıldığından, federe öğrenme ile artık geçerli değildir.

FL'ye yönelik saldırılar ele alındığında, verilerin anonimleştirilmesi üyelik çıkarımı, veri ve model zehirleme saldırılarından korunmak için bir adım daha ileri giderken, kötü niyetli istemcilerin ve yerel modellerin ortaya çıkardıkları birçok hata oranına dayalı güncelleme reddetmelerinin belirlenmesine de odaklanılmaktadır. Her iki saldırıda da saldırganın yerel modellere eğitim verisi ekleme ya da bu verilere müdahale etme imkanı olduğu varsayılmaktadır. Çalıştığımız Cross-Silo Federated Learning çerçevesinde bu saldırıların uygulanması zordur çünkü bu tür FL genellikle eğitimin güvenli ve kontrollü ortamlarda gerçekleştiği büyük kuruluşlar ve hastaneler tarafından kullanılır.

8. Sonuç ve Gelecek Çalışma

8.1 Tez Sonuç

Bu tezde, anonimleştirmenin makine öğrenimi model performansı üzerindeki etkilerini araştırdık ve manuel olarak oluşturulmuş genelleştirme hiyerarşi ağaçlarına ihtiyaç duymadan herhangi bir veri kümesi üzerinde çalışabilen otomatik bir genelleştirme çerçevesi geliştirmeye odaklandık. Eğitimden önce makine öğrenimi modelinin performansı hakkında faydalı bilgiler sağlayan yeni bir yaklaşım olan bilgi kaybı metriğini kullandık. Mondrian algoritmasını k-anonimlik çerçevemizi uygulamak için seçtik çünkü genel olarak en iyi performans gösteren algoritma olduğu bildirilmiştir [15]. Ayrıca, her bir uzantının farklı bir gizlilik düzeyi sunduğunu göz önünde bulundurarak, her birinde anonimleştirme davranışını anlamak için iki farklı k-anonimlik uzantısı ile çalıştık. Üç farklı veri kümesi üzerinde test ettik, her uzantı için farklı koşul değerlerini denedik ve sonuçlarımızı tartıştık.

Sonuçlarımız, ML model performansı söz konusu olduğunda standart *k-anonimliğin* en iyi performansı gösterdiğini, bunu l-çeşitlilik ve *t-yakınlığın* izlediğini göstermektedir. Ayrıca sonuçların farklı koşul değerlerine göre değişebildiğini de fark ettik. Test ettiğimiz ilginç bir durum, anonimleştirme sürecine bir optimizasyon adımı eklemektir; bu, gizliliğinden ödün vermeden standart *k-anonimlik* performansını artırabileceğini göstermiştir. Anonimleştirme süreci katı kısıtlamalara tabi tutulduğunda, ML modellerinin performansı genel olarak düşer; ancak, bozulma derecesi *k-anonimlik* uzantısına, koşul değerine, genelleştirmeye ve bastırma seçimlerine bağlıdır. Konut veri kümesinden elde ettiğimiz bulgular, bazı veri kümesi sütunlarının, özellikle de bu sütunlar hedef değerlerle yüksek korelasyona sahip olduğunda, ML performansını önemli ölçüde düşürmeden anonimleştirilemeyeceğini göstermektedir.

Ayrıca veri anonimleştirme, federe öğrenme ve bunların veri gizliliğindeki rolünü ayrıntılı olarak inceledik. Her iki yaklaşıma yönelik tehditleri tartıştık. Veri gizliliğini artıran ve sınıflandırma görevleri için minimum bilgi kaybı sağlayan ***k-anonimlik*** uyarlamalı bir çapraz silo federasyonlu öğrenme çerçevesi tanımladık. Önerilen çerçevemiz her iki

yaklaşımdan da faydalanmaktadır

yaklaşımlarının avantajlarından yararlanır ve her bir yaklaşımın tek başına savunmasız olduğu veri gizliliğini tehdit eden saldırılara karşı daha iyi koruma sağlar. Anonimleştirme tekniklerinin uygun şekilde entegre edilmesiyle minimum bilgi kaybı elde edilirken gizliliğin korunması açısından federe öğrenme performansının geliştirilebileceğini gösterdik. Federe öğrenme çerçevelerinde anonimleştirme etkilerinin araştırılmasına çok az odaklanıldığını fark ettik, bu nedenle Federe öğrenme uygulamamızı ayrıntılı olarak tartıştık. Bu çalışmanın gelecekte yapılacak daha fazla çalışma için bir başlangıç noktası olmasını umuyoruz.

8.2 Gelecek Çalışmalar Yönler

Anonimleştirme çerçevesi için gelecekteki çalışmalar, yeni gerçek hayat veri kümeleri üzerinde kapsamlı bir test içerebilir. Farklı dağılımlara ve çeşitli veri türlerine sahip 3 farklı veri kümesi üzerinde test yapmış olsak da, bu tür bir çalışma için uygun olan halka açık gerçek hayat veri kümelerinin genel bir eksikliği vardır. Ayrıca, bu çalışmanın bir uzantısı da k-anonimleştirilmiş veri kümeleri ile eğitilmiş ML modelleri ve FL modellerinin işlem süresi, gizlilik seviyesi ve performans açısından karşılaştırılması olabilir.

FL çerçevemiz için, FL'de kategorik veri kümeleri için çalışma ve uygulama eksikliği olduğunu fark ettik, bulduğumuz çoğu çalışma FL'yi görüntü tabanlı sınıflandırma görevleri için kullanıyor. Bu nedenle, FL'nin rolünü daha fazla araştırmak için daha fazla veri kümesi üzerinde test yapmak hayati önem taşımaktadır. Bazı FL türleri için, kötü niyetli istemcilere karşı daha iyi koruma sağlamak amacıyla istemci seçimlerine yönelik daha fazla stratejinin araştırılması bir sonraki adım olacaktır.

REFERANSLAR

[1] Avrupa Birliği, "Kişisel verilerin işlenmesine ilişkin olarak gerçek kişilerin korunması ve bu tür verilerin serbest dolaşımına ilişkin ve 95/46/ec sayılı direktifi (genel veri koruma tüzüğü) yürürlükten kaldıran 27 Nisan 2016 tarihli ve 2016/679 sayılı Avrupa Parlamentosu ve Konsey Tüzüğü (eu)," s. 1-88, 2016. [Çevrimiçi]. Mevcut: http://data.europa.eu/eli/reg/2016/679/oj

[2] Türkiye, "Yönetmelik (tr) 2016/6698," 2016.[Çevrimiçi]. Mevcut: https://www.kvkk.gov.tr/Icerik/6649/Personal-Data-Protection-Law

[3] A. Majeed and S. Lee, "Anonymization techniques for privacy preserving data publishing: A comprehensive survey," IEEE Access, vol. 9, pp. 8512-8545, 2021.

[4] P. Samarati ve L. Sweeney, "Bilgi ifşa ederken mahremiyeti korumak: ***k***-anonimlik ve bunun genelleme ve bastırma yoluyla uygulanması," Tech. Rep., 1998.

[5] A. Gionis ve T. Tassa, "k-anonymization with minimal loss of information," IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 2, pp. 206-219, 2009.

[6] A. Machanavajjhala, J. Gehrke, D. Kifer, ve M. Venkitasubramaniam, "L-diversity: privacy beyond ***k-anonymity,***" in 22nd International Conference on Data Engineering (ICDE'06), 2006, pp. 24-24.

[7] N. Li, T. Li ve S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in 2007 IEEE 23. Uluslararası Veri Mühendisliği Konferansı, 2007, pp. 106-115.

[8] L. Sweeney, "Achieving ***k-anonymity*** privacy protection using generalization and suppression," Int. J. Uncertain. Fuzziness Knowl. Based Syst., vol. 10, pp. 571-588, 2002.

[9] C. C. Aggarwal ve P. S. Yu, A General Survey of Privacy- Preserving Data Mining Models and Algorithms. Boston, MA: Springer US, 2008, s. 11-52.

[10] T. De Waal ve L. Willenborg, "Küresel yeniden kodlama ve yerel bastırma yoluyla bilgi kaybı," 01 1999.

[11] J.-W. Byun, A. Kamra, E. Bertino ve N. Li, "Efficient ***k***- anonymization using clustering techniques," in Advances in Databases: Concepts, Systems and Applications, R. Kotagiri, P. R. Krishna, M. Mohania, and E. Nantajeewarawat, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, s. 188- 200.

[12] B. Malle, P. Kieseberg, and A. Holzinger, "Interactive anonymization for privacy aware machine learning," in IAL@PKDD/ECML, 2017.

[13] S. Shaham, M. Ding, B. Liu, S. Dang, Z. Lin ve J. Li, "Privacy preserving location data publishing: A machine learn- ing approach," IEEE Transactions on Knowledge and Data Engineering, vol. 33, pp. 3270-3283, 2021.

[14] F. Liu ve T. Li, "A clustering k-anonymity privacy-preserving method for wearable iot devices," Secur. Commun. Networks, vol. 2018, pp. 4 945 152:1-4 945 152:8, 2018.

[15] W. Mahanan, W. A. Chaovalitwongse ve J. Natwichai, "Data anonymization: a novel optimal ***k-anonymity*** algorithm for identical generalization hierarchy data in iot," Serv. Oriented Comput. Appl., vol. 14, no. 2, pp. 89-100, 2020.

[16] R. Khan, X. Tao, A. Anjum, T. Kanwal, S. u. R. Malik, A. Khan, W. u. Rehman, and C. Maple, "-sensitive ***k***- anonymity: An anonymization model for iot based electronic health records," Electronics, vol. 9, no. 5, 2020.

[17] H. Wimmer ve L. M. Powell, "A comparison of the effects of k-anonymity on machine learning algorithms," International Journal of Advanced Computer Science and Applications, vol. 5, pp. 155-160, 2014.

[18] M. Last, T. Tassa, A. Zhmudyak ve E. Shmueli, "Improving accuracy of classification models induced from anonymized datasets," Information Sciences, vol. 256, pp. 138-161, 2014.

[19] D. Slijepcevic, M. Henzl, L. D. Klausner, T. Dam, P. Kieseberg ve M. Zeppelzauer, "***k-anonymity*** in practice: How generalisa- tion and suppression affect machine learning classifiers," ArXiv, vol. abs/2102.04763, 2021.

[20] B. C. Fung, K. Wang ve P. S. Yu, "Anonymizing classification data for privacy preservation," IEEE Transactions on Knowl- edge and Data Engineering, vol. 19, no. 5, pp. 711-725, 2007.

[21] K. LeFevre, D. DeWitt, ve R. Ramakrishnan, "Mondrian multidimensional ***k***-anonymity," in 22nd International Conference on Data Engineering (ICDE'06), 2006, pp. 25-25.

[22] N. Mohammed, B. C. Fung, P. C. Hung, ve C.-k. Lee, "Sağlık verilerinin anonimleştirilmesi: A case study on the blood transfusion service," in Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '09. New York, NY, ABD: Association for Computing Machinery, 2009, s. 1285-1294.

[23] J. Goldberger ve T. Tassa, "Efficient anonymizations with enhanced utility," in 2009 IEEE International Conference on Data Mining Workshops, 2009, pp. 106-113.

[24] K. El Emam, F. K. Dankar, R. Issa, E. Jonker, D. Amyot, E. Cogo, J.-P. Corriveau, M. Walker, S. Chowdhury, R. Vaillancourt, T. Roffey, ve J. Bottomley, "A Globally Optimal ***k***-Anonymity Method for the De-Identification of Health Data," Journal of the American Medical Informatics Association, vol. 16, no. 5, pp. 670-682, 09 2009.

[25] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, ve A. W.- C. Fu, "Utility-based anonymization using local recoding," in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '06. New York, NY, ABD: Association for Computing Machinery, 2006, s. 785-790.

[26] J.-L. Lin ve M.-C. Wei, "An efficient clustering method for ***k-anonymization,***" ser. PAIS '08. New York, NY, ABD: Association for Computing Machinery, 2008, s. 46-50.

[27] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.

[28] P. Geurts, "Extremely randomized trees," in MACHINE LEARN- ING, 2003, s. 2006.

[29] N. Cristianini ve J. Shawe-Taylor, "An introduction to support vector machines and other kernel-based learning methods," 2000.

[30] T. Chen ve C. Guestrin, "Xgboost: Ölçeklenebilir bir ağaç güçlendirme sistemi." Teknoloji. Rep., 2016.

[31] E. Wilson ve D. Tufts, "Multilayer perceptron design algorithm," in Proceedings of IEEE Workshop on Neural Networks for Signal Processing, 1994, pp. 61-68.

[32] Avrupa Birliği, "Kişisel verilerin işlenmesine ilişkin olarak gerçek kişilerin korunması ve bu tür verilerin serbest dolaşımına ilişkin ve 95/46/ec sayılı direktifi (genel veri koruma tüzüğü) yürürlükten kaldıran 27 Nisan 2016 tarihli ve 2016/679 sayılı Avrupa Parlamentosu ve Konsey Tüzüğü (eu)," s. 1-88, 2016.

[33] P. Samarati ve L. Sweeney, "Bilgi ifşa ederken mahremiyeti korumak: ***k***-anonimlik ve bunun genelleme ve bastırma yoluyla uygulanması," Tech. Rep., 1998.

[34] I. Buratovi´c, M. Mili˘cevi´c, ve K. ˘Zubrini´c, "Effects of data anonymization on the data mining results," in 2012 Proceedings of the 35th International Convention MIPRO, 2012, pp. 1619- 1623.

[35] J. Koneˇcn´y, H. B. McMahan, F. X. Yu, P. Richtarik, A. T. Suresh, ve D. Bacon, "Federated learning: Strategies for improving communication efficiency," in NIPS Workshop on Private Multi-Party Machine Learning, 2016.

[36] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. A. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D'Oliveira, S. E. Rouayheb, D. S. Zhao, "Advances and open problems in federated learning," CoRR, vol. abs/1912.04977, 2019.

[37] Q. Yang, Y. Liu, T. Chen ve Y. Tong, "Federated machine learning: Concept and applications," arXiv: Artificial Intelligence, 2019.

[38] P. M. Mammen, "Federated learning: Opportunities and challenges," ArXiv, vol. abs/2101.05428, 2021.

[39] K. Nandury, A. Mohan ve F. Weber, "Çeşitlilik ölçeklendirme ve yarı denetimli öğrenme ile bulutta çapraz silo federasyon eğitimi," ICASSP 2021 - 2021 IEEE International Confer- ence on Acoustics, Speech and Signal Processing (ICASSP), 2021, s. 3085-3089.

[40] A. Durrant, M. Markovic, D. Matthews, D. May, J. A. Enright ve G. Leontidis, "The role of cross-silo federated learning in facilitating data sharing in the agri-food sector," CoRR, vol. abs/2104.07468, 2021. [Çevrimiçi]. Mevcut: https: //arxiv.org/abs/2104.07468

[41] C. Che, X. Li, C. Chen, X. He ve Z. Zheng, "A decentralized federated learning framework via committee mechanism with convergence guarantee," CoRR, vol. abs/2108.00365, 2021.

[42] R. Shokri, M. Stronati, C. Song ve V. Shmatikov, "Member- ship inference attacks against machine learning models," 2017 IEEE Symposium on Security and Privacy (SP), s. 3-18, 2017.

[43] A. Machanavajjhala, J. Gehrke, D. Kifer, ve M. Venkitasubramaniam, "L-diversity: privacy beyond ***k-anonymity,***" in 22nd International Conference on Data Engineering (ICDE'06), 2006.

[44] N. Li, T. Li ve S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in 2007 IEEE 23rd Inter- national Conference on Data Engineering, 2007, pp. 106-115.

[45] S. Yoo, M. Shin, and D. Lee, "An approach to reducing infor- mation loss and achieving diversity of sensitive attributes in ***k***- anonymity methods," Interactive Journal of Medical Research, vol. 1, 2012.

[46] K. LeFevre, D. DeWitt, and R. Ramakrishnan, "Mondrian multidimensional ***k***-anonymity," in 22nd International Conference on Data Engineering (ICDE'06), 2006.

[47] D. Slijepcevic, M. Henzl, L. D. Klausner, T. Dam, P. Kieseberg ve M. Zeppelzauer, "Uygulamada ***k-anonimlik:*** How generalisation and suppression affect machine learning classifiers," CoRR, vol. abs/2102.04763, 2021.

[48] S. Ruder, "An overview of gradient descent optimization algorithms," ArXiv, vol. abs/1609.04747, 2016. [18] H. B. McMahan, E. Moore, D. Ramage, S. Hampson ve B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in AISTATS, 2017.

[49] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, ve B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in AISTATS, 2017.