T.C.
VAN YÜZÜNCÜ YIL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
İSTATİSTİK ANABİLİM DALI

ÇEKİŞMELİ ÜRETİCİ AĞLAR (ÇÜA) İLE SENTETİK VERİ ÜRETME

YÜKSEK LİSANS TEZİ

HAZIRLAYAN: Hayrullah URCAN
DANIŞMAN: Dr.Öğr. Üyesi Murat CANAYAZ

VAN-2022

T.C.
VAN YÜZÜNCÜ YIL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
İSTATİSTİK ANABİLİM DALI

ÇEKİŞMELİ ÜRETİCİ AĞLAR (ÇÜA) İLE SENTETİK VERİ ÜRETME

YÜKSEK LİSANS TEZİ

HAZIRLAYAN: Hayrullah URCAN

VAN-2022

KABUL VE ONAY SAYFASI

İstatistik Anabilim Dalı'nda Dr. Öğ. Üyesi Murat CANAYAZ danışmanlığında, Hayrullah URCAN tarafından sunulan "Çekişmeli Üretici Ağlar (ÇÜA) İle Sentetik Veri Üretme " isimli bu çalışma Lisansüstü Eğitim ve Öğretim Yönetmeliği'nin ilgili hükümleri gereğince …/...../..... Tarihinde aşağıdaki jüri tarafından oy birliği / oy çokluğu ile başarılı bulunmuş ve Yüksek Lisans tezi olarak kabul edilmiştir.

Başkan:..... İmza:

Üye:..... İmza:

Üye:..... İmza:

Üye:..... İmza:

Üye:..... İmza:

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun …./..../…... tarih ve ..... Sayılı kararı ile onaylanmıştır.

İmza

……..……..

Enstitü Müdürü

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

Hayrullah URCAN

ÖZET

ÇEKİŞMELİ ÜRETİCİ AĞLAR (ÇÜA) İLE SENTETİK VERİ ÜRETME

URCAN, Hayrullah
Yüksek Lisans Tezi, İstatistik Anabilim Dalı
Tez Danışmanı: Dr. Öğr. Üyesi Murat CANAYAZ
Ocak 2022, 46 Sayfa

Çekişmeli üretici ağlar olarak adlandırılan üretici modeller, klasik derin ağ mimarilerinden farklı olarak, üretici ağlar (Ü) ve ayırt edici ağlar (A) olmak üzere iki farklı derin ağa sahiptir. Bu iki ağın çekişmesi ile öğrenme işlemi gerçekleştirilmektedir. ÇÜA modelleri, sentetik görüntülerin ve metinlerin oluşturulmasında büyük olanaklar sağlamıştır. Son birkaç yılda, geleneksel istatistiksel tekniklerin aksine veri dağılımlarını modelledikleri, büyük esneklik sundukları ve tablo halinde veri ürettikleri için ÇÜA'lar kullanılmaya başlanmıştır.

Bu çalışmada tablo verilerinin modellenmesiyle ilgili olarak, TGAN, CTGAN ve CopulaGAN gibi çeşitli modellerini kullanılmaktadır. ÇÜA'lar sentetik veri üretimi için klasik yöntemlerden daha iyi sonuçlar göstermiştir. Spesifik olarak, modeller, bu yöntemin ihtiyaç duyduğu sınırlamalar ve gereksinimler göz önünde bulundurularak deprem veri setinin ÇÜA modelleri üzerinden eğitilmektedir. Üretilen sentetik veriler görsel, istatistiksel ve makine öğrenimi tabanlı olarak değerlendirilmektedir.

Anahtar kelimeler: ÇÜA, Deprem veri seti, Sentetik veri seti, Tablo veri üretimi.

ABSTRACT

SYNTHETIC DATA GENERATION WITH COMPETITIVE GENERATIVE ADVERSARIAL NETWORKS (GAN)

URCAN, Hayrullah
M.Sc. Thesis, Department of Statistics
Supervisor : Asst. Prof. Dr. Murat CANAYAZ
January 2022, 46 Pages

Generator models, called contentious generator networks, have two different deep networks, namely generator networks (G) and distinctive networks (D), unlike classical deep network architectures. It performs the learning process by the contention of these two networks. GAN models have opened up great possibilities in the creation of synthetic images and text. In the last few years, GANs have started to be used because they model data distributions, offer great flexibility, and produce tabular data, unlike traditional statistical techniques.

In this study, various algorithms such as TGAN, CTGAN and CopulaGAN are used for modeling tabular data. GANs showed better results than classical methods for synthetic data generation. Specifically, the models are trained on the GAN models of the earthquake dataset, taking into account the limitations and requirements of this method. The synthetic data produced are evaluated visually, statistically and based on machine learning.

Keywords: GAN, Earthquake dataset, Synthetic dataset, Table data generation.

ÖN SÖZ

Bu tez çalışmasında, her türlü ilgi ve yardımlarını esirgemeyen danışmanım sayın Dr. Öğr. Üyesi Murat CANAYAZ'a, büyük emeklerinden dolayı teşekkür ederim.

Ayrıca, desteği ile her zaman yanımda olan değerli eşim sayın Ferayi GÜZEL URCAN'a, oğlum Ahmet Ömür'e teşekkür ederim.

2022

Hayrullah URCAN

İÇİNDEKİLER

Sayfa

Sayfa

ÇİZELGELER LİSTESİ

Çizelge **Sayfa**

ŞEKİLLER LİSTESİ

Şekil **Sayfa**

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış kısaltmalar açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar	Açıklama
CDF	Kümülatif Dağıtım İşlevi
CopulaGAN	Kopula Çekişmeli Üretici Ağlar
CTGAN	Koşullu Tablo Çekişmeli Üretici Ağlar
ÇÜA	Çekişmeli Üretici Ağlar
ESK	Elektronik Sağlık Kayıtları
GAN	Generative Adversarial Networks
GKM	Gauss Karışım Modelli
PCA	Temel Bileşen Analizi
PDF	Olasılık Yoğunluk Fonksiyonu
RDT	Tersinir Veri Dönüşümü
ROC AUC	Performans Ölçümü
SDV	Açık Kaynak Veri Kütüphanesi
SNN	Simüle Edilmiş Sinir Ağları
TGAN	Tabular Çekişmeli Üretici Ağlar
UKSB	Uzun Kısa Süreli Bellek
VOK	Varyasyonlu Otomatik Kodlayıcı
YSA	Yapay sinir ağları

1. GİRİŞ

Son yıllarda derin öğrenme teknikleri yapay öğrenme alanında devrim yaratmıştır (Teng, 2017). Derin öğrenme, gerçek dünya veri kümeleri üzerinde karmaşık yapıların öğrenildiği bir makine öğrenimi yaklaşımı olarak ortaya çıkmıştır. Büyük ölçekli veriler kullanılarak eğitilen derin sinir ağları, görüntülerin anlamsal olarak anlaşılması açısından klasik görüntü işleme tekniklerini önemli ölçüde geride bırakmıştır (Wason, 2018).

Yapay zekanın yaygınlaşmasıyla birlikte, son birkaç yılda makine öğrenimi yöntemleri, saldırı tespit sistemlerinde sağlam ve doğru bir savunma mekanizması olarak uygulanmakta ve dikkate değer bir performans sergilemektedir. Özellikle, karar ağaçları, yapay sinir ağları (makine öğrenimi, derin öğrenme), evrişimli sinir ağları destek vektör makineleri ve bayes ağları gibi çeşitli makine öğrenimi sınıflandırıcıları saldırı tespit sistemlerinde yaygın olarak kullanılmaktadır. Bu sınıflandırıcılar geleneksel yöntemlerin aksine, şüpheli trafiği tespit edebilir veya verilerdeki görünmeyen kalıpları ve anormallikleri keşfedebilmektedir (Li ve ark., 2017; Teng ve ark., 2018).

Günümüzde kuruluşlar, genellikle insanlar tarafından yürütülen süreçleri ve iş akışlarını akıllı bir şekilde artırmak için ilişkisel tablo verileri üzerinde makine öğrenimini giderek daha fazla kullanmaktadır. Veri bilimi platformu KAGGLE tarafından gerçekleştirilen yakın tarihli bir araştırmaya göre, tablo verileri iş dünyasında en sık karşılaşılan veri türü ve akademide en yaygın kullanılan ikinci formattır (Latif ve ark., 2020). Aynı zamanda, araştırmacılar, kritik darboğazlar çözmek (Che ve ark., 2017; Patki ve ark., 2016), veri erişiminde karşılaşılan bürokratik engelleri ortadan kaldırmak ve "güvenli veri alanı" sağlamak da dahil olmak üzere bir dizi yaygın veri bilimi endişesini hafifletme yeteneği için sentetik verileri oluşturmaya başlamıştır (Choi ve ark., 2017; Chandar ve ark., 2016). Sentetik veri kümeleri, yeni araçların performanslarını test etmek veya bunları kullanarak kendi kendini süren arabalar geliştirmek gibi belirli ihtiyaçlara uyacak şekilde oluşturulabilmekte veya veri paylaşırken istenmeyen beklenmedik durumları ortadan kaldırabilmektedir. Örneğin, bir şirket çalışanlarına, arkadaşlarına veya ünlülere ait olabilecek verilere erişmelerini önlemek için sentetik veriler verebilir veya yanlışlıkla bir ihlal durumunda riski ortadan kaldırmak için harici danışmanlara sentetik veriler sağlayabilmektedir.

Sentetik veri üretimi, belirli bir veri kümesi için ortak birçok değişkenli olasılık dağılımının modellenmesi ve ardından bu dağılımdan örnekleme yoluyla gerçekleştirilir. Karmaşık veri kümeleri daha karmaşık dağılımlar gerektirir. Örneğin, bir dizi olay gizli Markov modelleri kullanılarak modellenmiş olabilir veya bir dizi doğrusal olmayan bağıntılı değişkenler kopulalar kullanılarak modellenebilmektedir. Bununla birlikte, bu üretici modeller, kullanıcılar için mevcut olan dağıtım fonksiyonlarının türü ile kısıtlanır, bu da üretici modeller oluşturmak için kullanılabilecek temsilleri ciddi şekilde sınırlar ve ardından sentetik verilerin aslına uygunluğunu sınırlamaktadır.

Veri sentezinin temel işlevi, daha önce hiç görülmemiş yeni veriler üretmektir. Bu, veri bilimcileri tarafından model değerlendirmesi ve eğitimi için ayrı bir veri kümesi olarak veya yazılım mühendisleri için test verileri olarak kullanabilmektedir. Bu yeni veri kümeleri, gerçek veri noktalarından hiçbirini içermediğinden, bu yöntem kullanıcıların kendi başına birçok gizlilik avantajına sahip olan gerçek verilere erişmesine izin vermemektedir. Ayrıca, bu veriler mevcut verilere ek olarak kullanılmaktadır. Mevcut derin öğrenme modelleri, iyi bir genelleme için genellikle çok fazla veri gerektirmektedir. Bir veri setinde 10.000 satır veri kümesi olduğunu varsayalım, ancak etkili genelleme için on kat daha fazlasına ihtiyaç vardır. Gerçekçi sentetik veriler daha sonra normalde mevcut olmayacak modellerin eğitimine izin vermektedir (Brenninkmeijer ve ark., 2019).

Aynı zamanda, istatistiksel bilimler topluluğundaki bazı araştırmacılar, sentetik veriler oluşturmak için rastgeleliğe dayalı yöntemler kullanmaya başlamıştır (Reiter, 2005; Kinney ve ark., 2011). Bu çabaların çoğu, verilerin ifşa edilmesini ve verilerle temsil edilen kişilerin (genellikle bir ankete katılanların) mahremiyetini korumayı amaçlamıştır (Goodfellow ve ark., 2016). Varyasyonlu otomatik kodlayıcılar ve daha sonra çekişmeli üretici ağları (ÇÜA) ve bunların sayısız uzantıları gibi nöral modeller kullanılarak üretken modellerin oluşturulması, hem veriyi temsil etmede sunulan performans hem de sunulan esneklik açısından çekicidir (Goodfellow ve ark., 2016).

Bu çalışmada, en umut verici gizliliği koruyan yaklaşımlardan biri sentetik veri üretimidir. Sentetik olarak oluşturulan veriler, gizlilik endişeleri olmadan herkese açık olarak paylaşılabilir ve tahmin modelleri oluşturma ve kalıp bulma gibi görevler de dahil olmak üzere birçok ortak araştırma fırsatı sunmaktadır.

Sentetik veriler doğası gereği üretken bir süreci içerdiğinden, Çekişmeli Üretici Ağlar (ÇÜA'lar), sentetik veri üretimi için popüler bir seçim haline gelmiştir (Goodfellow ve

ark., 2014). Günümüzde daha kararlı ve gerçekçi örnekler üretme yeteneğine sahip çeşitli ÇÜA türleri mevcuttur. Bu çalışmada, tablo veri üretimi için ÇÜA mimarisi kullanılmaktadır. Kullanılan mimariler şunlardır.

- Tabular ÇÜA (TGAN)
- Koşullu Tablo ÇÜA (CTGAN)
- CopulaGAN.

2. KAYNAK BİLDİRİŞLERİ

Derin ağların ilk oluşturma modeli 2014 yılında Goodfellow tarafından tanıtılmıştır (Goodfellow ve ark., 2014). Çekişmeli üretici ağlar olarak adlandırılan üretici modeller, İngilizce yazılışının baş harfleriyle "GAN" olarak adlandırılmıştır. Klasik derin ağ mimarilerinden farklı olarak, üretici ağlar (Ü) ve ayırt edici ağlar (A) olmak üzere iki farklı derin ağa sahiptir. Bu iki ağın çekişmesi ile öğrenme işlemi gerçekleştirilmektedir. ÇÜA'lardaki gelişmeler ile bu ağların hızı ve eğitim performansı iyileştirilmiştir. Araştırmacılar tarafından, eğitim sırasında ağırlık kesmenin yan etkilerini ele almak için Wasserstein ÇÜA üzerinde gradyan cezasının getirilmesi önerilmiştir (Gulrajani, 2017). Bu çeşitli uygulamalar, özellikle görüntü oluşturmaya odaklanan, ÇÜA'lar için dikkate değer sonuçlar göstermiştir (Radford, 2016; Liu, ve ark., 2020). Sentetik veri örneklerin üretilmesi ile ilgili olarak, ÇÜA'lar bazı yöntemler de denenmiştir (Lin, 2018; Charlier, 2019). ÇÜA'ları kullanarak sentetik trafik verileri oluşturulmuştur. Bu tür modeller kötü amaçlı yazılım örneklerini sentezlemek için de kullanılmaktadır (Hu, 2017).

Ancak, bu verilerin sentezlenme şekli değişmektedir. Daha önce sentetik veriler, genellikle çok değişkenli olasılık dağılımı modellenerek oluşturulmuştur. Örnek modeller, Bayes ağlarını ve Gauss Copulas'ı içermektedir. Bazı araştırmacılar cevaplarını rastgeleye dayalı yöntemler kullanarak da bulmuşlardır. Ancak, bu yöntemlerin çoğu veri boyutu veya karmaşıklığı ile ilgili bir veya daha fazla kısıtlamaya sahiptir (Kinney ve ark., 2019). Son zamanlarda ÇÜA'lar üzerindeki çalışmalar, tablo verileriyle umut verici sonuçlar göstermiştir (Xu ve ark., 2018). ÇÜA'ların istatistiksel benzerlerinin daha iyi olup olmayacağı sorusu yükselmiş ve birkaç araştırmacı bunları yanıtlamaya çalışmıştır (Xu ve ark., 2018; Skoularidou ve ark., 2021). Araştırmacılar birden çok veri kümesindeki ÇÜA'ları çeşitli istatistiksel yöntemlerle karşılaştırmış ve ÇÜA'ların çoğu görevde bu klasik yöntemlerden daha iyi performans gösterdiğini saptamıştır (Xu ve ark., 2018).

Tablo verilerinin modellenmesiyle ilgili olarak, TGAN, CTGAN ve CopulaGAN gibi çeşitli algoritmalar, ÇÜA'ların tablolu sentetik veri üretimi için klasik yöntemlerden daha iyi performans gösterdiği kanıtlanmıştır (Xu ve ark., 2018; Xu ve ark., 2019). Bu

nedenle tablo verilerini işlemek için birkaç ÇÜA modeli kullanılmıştır. Kunar ve arkadaşları, karmaşık dağılımlarla çeşitli veri türlerini modelleyebilen koşullu bir tablo ÇÜA olan CTAB-GAN 'ı tanıtmıştır (Zhao ve ark., 2021; Mottini ve ark., 2018). Yolcu kayıt bilgilerinden oluşan veri setini, MedGAN kategorik özellik yerleştirme ve bir Çapraz Ağ mimarisi kullanarak sentezlemişlerdir. Araştırmacılar, Elektronik Sağlık Kayıtları (ESK) verileri üzerinde sürekli zaman serileri oluşturmak için ÇÜA'ları kullanırken, diğer araştırmacılar, bir otomatik kodlayıcıyı bir ÇÜA ile birleştiren MedGAN'ın ESK verilerinden yüksek boyutlu ayrık değişkenler üretmesi önermiştir (Choi ve ark., 2021). TGAN, sentetik verilerin anlamsal bütünlüğünü artırmak için bir evrişimsel ayırıcı ve bir evrişimsiz oluşturucu ve bir sınıflandırıcıdan oluşmaktadır (Park ve ark., 2018). Tablo verilerini sentezlemek için koşullu bir üreteç kullanan CTGAN önerilmiştir (Xu ve ark., 2018). CopulaGAN, CTGAN model eğitimini kolaylaştırmak için Kümülatif Dağılım Fonksiyonu (SDV) tabanlı dönüşümü kullanan TGAN modelinin bir varyasyonu olarak kullanılmıştır (Patki ve ark., 2018).

Sentetik veri kasası (SDV), kullanıcıların daha sonra orijinal veri kümesiyle aynı biçime ve istatistiksel özelliklere sahip yeni sentetik veriler oluşturmak için tek tablolu, çoklu tablolu ve zaman serisi veri kümelerini kolayca öğrenmesine olanak tanıyan bir sentetik veri oluşturma kütüphanesidir (Patki ve ark., 2016). SDV'de tek değişkenli marjinaller bir Gauss karışım modeli kullanılarak çok değişkenli bir Gauss kopulası olarak öğrenilmektedir. Kopula Bayes ağları koşullu kopulaları kullanmakta ve değişkenler üzerinden bir Bayes ağı öğrenmektedir (Elidan, 2010). Araştırmacılar, ağ yapısını bulmak için Monte-Carlo ağaç aramasını kullanmayı önermişlerdir (Chang ve ark., 2019).

ÇÜA tabanlı yöntemler, sentetik veri üretimindeki son ilerlemelerin çoğunun yürütücülüğünü üstlenmiştir. TGAN ve CTGAN karışık değişken türlerini ele almak için koşullu bir üreteç oluşturmakta, CopulaGAN Kümülatif Dağıtım İşlevi tabanlı dönüşümden yararlanmaktadır (Patki ve ark., 2018). Tıbbi kayıtlar için veri oluşturuculardа ÇÜA'lara dayanmaktadır (Che ve ark., 2017). Tıbbi görüntülerin sentezlenmesi, çoğunlukla tıbbi verilere bağlı gizlilik düzenlemeleri nedeniyle ÇÜA'ların kullanılabileceği en önemli alanlardandır (Yi ve ark., 2018). ÇÜA'lar, görüntülerin normal çevirisinden ziyade görüntü dağılımını öğrenerek veri eksikliğine daha genel bir çözüm sunmaktadır (Yi ve ark., 2018).

3. MATERYAL VE YÖNTEM

3.1. Makine Öğrenimi

Makine öğrenimi, büyük miktarda verideki kuralları tespit etmek için yararlı olan bir yapay zeka alanıdır. Algılanan kurallar kullanarak, makine öğrenme algoritması gelecekteki verileri tahmin edebilir veya belirsizlik altında karar verebilmektedir (Murphy, 2012). Makine öğrenimi, temel olarak, bilgisayar algoritmalarının bir tür belirsizliğin söz konusu olduğu karmaşık işlevleri istatistiksel olarak tahmin etmeyi öğrendiği bir uygulamalı istatistik biçimidir (Goodfellow, 2016). Spam filtreleri, el yazısı tanıma ve diğerlerinin yanı sıra yüz tanıma gibi farklı türde algoritmalar günümüzde yaygın olarak kullanılmaktadır (Murphy, 2012).

Makine öğrenimi algoritmaları genellikle iki kategoriye ayrılmaktadır: denetimli öğrenme ve denetimsiz öğrenme, burada denetimli öğrenme pratikte en yaygın kullanılan biçimdir (Murphy, 2012). Denetimli öğrenmede amaç, bir dizi etiketli girdi-çıktı çifti verilen x girdilerinden z çıktılarına bir eşleme öğrenmektir (Murphy, 2012). Her çift, bir x_i girdisinden ve z_i adlı bir çıktı etiketinden oluşmaktadır; burada etiket, örneğin girdinin ait olduğu bir sınıfı temsil edebilmektedir. Girdiler, işlenmekte olan verilerden gerçek özellikleri temsil eden bir dizi özellikten oluşmaktadır (Murphy, 2012). En yaygın denetimli öğrenme türü, algoritmanın amacının bir dizi x girdisi alıp her girdi için doğru sınıf etiketlerini z'ye döndürdüğü sınıflandırmadır (Murphy, 2012). Doğru etiket, giriş-çıkış çiftleri ile eğitim yoluyla öğrenilen bir fonksiyondan alınan her bir girişi genelleştirerek atanmaktadır (Goodfellow, 2016).

Bir çiçeğin belirli bir türe ait olup olmadığını sınıflandırmak için, makine öğrenimi algoritmasının girdisi taç yaprağı uzunluğu, taç yaprağı genişliği veya çiçeğin diğer özellikleri olabilir ve etiket belirli türleri temsil etmektedir. Algoritma, belirli özellikleri ve ait oldukları ilgili türleri içeren veriler üzerinde eğitilmektedir.Daha sonra özellikleri doğru sınıflara eşlemek için bir fonksiyon öğrenilmektedir. Eğitimden sonra, algoritma girdi özelliklerini alır ve eğitimden öğrenilen fonksiyon ile her girdi için bir sınıf tahmin etmektedir.

Makine öğrenmesi algoritmasının öğrenebilmesi için kendi performansını değerlendirebilmesi gerekmektedir. Performansı değerlendirmek için kullanılan yöntem genellikle algoritmanın gerçekleştirmek üzere ayarlandığı belirli göreve bağlıdır (Goodfellow, 2016). Sınıflandırma için genellikle modelin doğruluğu kullanılır ve algoritmanın tüm girdilerden doğru olarak sınıflandırdığı girdi miktarı olarak hesaplanmaktadır (Goodfellow, 2016). Algoritmanın performansı, bir hata oranı olarak da ölçülebilmektedir. Algoritmanın hatalı çıktı verdiği girdilerin, doğruluğun tersi olan tüm girdiler oranı olarak kabul edilmektedir (Goodfellow, 2016). Performans ölçümüne genellikle kayıp fonksiyonu denir. Bu kayıp fonksiyonu, bir optimizasyon algoritması tarafından gerçekleştirilen algoritmayı mümkün olduğunca iyi performans gösterecek şekilde optimize etmek için kullanmaktadır (Goodfellow, 2016).

3.2. Derin Öğrenme

Derin öğrenme terimi genellikle birkaç veya daha fazla gizli katmana sahip sinir ağları ifade edilmektedir. 2012 yılında AlexNet'in tanıtılmasıyla ortaya çıkmıştır (Krizhevsky ve ark., 2012). AlexNet, ImageNet yarışmasını (ImageNet Büyük Ölçekli Görsel Tanıma Yarışması) önemli bir farkla kazanan derin bir evrişimsel sinir ağıdır. O zamandan beri sinir ağları, yapay zeka alanında en çok araştırılan konulardan biri haline gelmiştir. Esasen her şeyi modelleme kapasitesine sahip olan sinir ağları, yalnızca bilgisayar alanında değil, aynı zamanda doğal dil işlemede, konuşma işlemede ve özerk yerlerde de devrim yaratmıştır.

Derin üretici modeller alanı, sırasıyla 2013 ve 2014'te Varyasyonlu Otomatik Kodlayıcı (VOK) ve Çekişmeli Üretici Ağ (ÇÜA) ile birlikte derin öğrenmeden bile daha yenidir (Kingma ve Welling., 2019; Goodfellow ve ark., 2014). Standart bir otomatik kodlayıcı şekil 3.1’de gösterilmektedir.

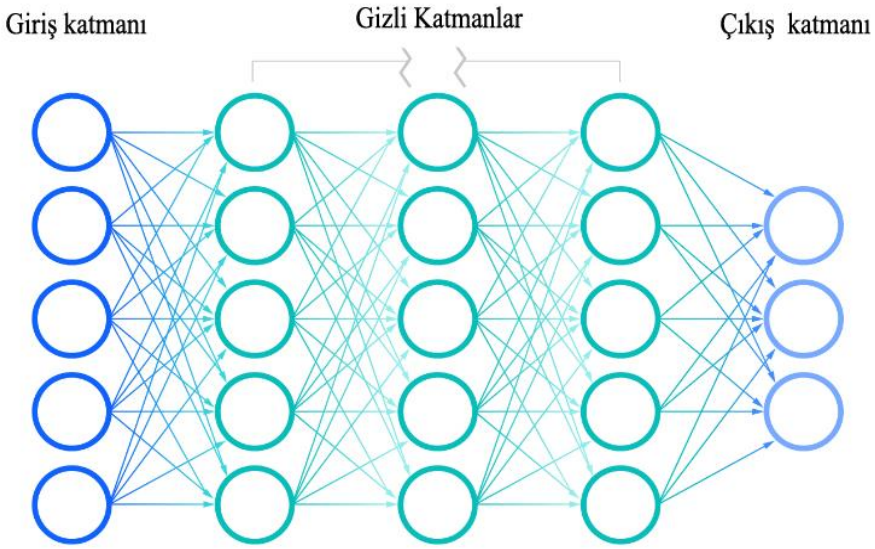
Şekil 3.1. Standart bir otomatik kodlayıcıya genel bakış.

Derin üretici modeller, verilerin dağılımını takiben örnekler üretebilen çok katmanlı sinir ağı ile karakterize edilmektedir. En önemli fark, Varyasyonlu Otomatik Kodlayıcının bu dağılımı açıkça modellemeye çalışmasıdır, bu da yaklaşık sonsuz çıkarım yapabilme avantajına sahiptir. Yani bazı x verileri verildiğinde, bu x'e neden olan gizli yöntemi bulabilmektedir. Çekişmeli üretici ağlar ise bu özelliği taşımayan verileri örtük olarak modellemeye çalışmaktadır.

3.3. Yapay Sinir Ağları

Yapay sinir ağları (YSA) veya simüle edilmiş sinir ağları (SNN'ler) olarak da bilinen sinir ağları, makine öğreniminin bir alt kümesidir ve derin öğrenme algoritmalarının kalbinde yer almaktadır. Adları ve yapıları, biyolojik nöronların birbirine sinyal gönderme şeklini taklit ederek insan beyninden esinlenmiştir.

Yapay sinir ağları (YSA), bir girdi katmanı, bir veya daha fazla gizli katman ve bir çıktı katmanı içeren bir düğüm katmanından oluşmaktadır. Her düğüm veya yapay nöron diğerine bağlanır ve ilişkili bir ağırlık eşiğe sahiptir. Herhangi bir düğümün çıktısı belirtilen eşik değerinin üzerindeyse, o düğüm etkinleşmekte ve ağın bir sonraki katmanına veri gönderebilmektedir. Aksi takdirde, ağın bir sonraki katmanına hiçbir veri iletmemektedir (Gershenson 2003). Yapay sinir ağları grafiği şekil 3.2’de gösterilmektedir.

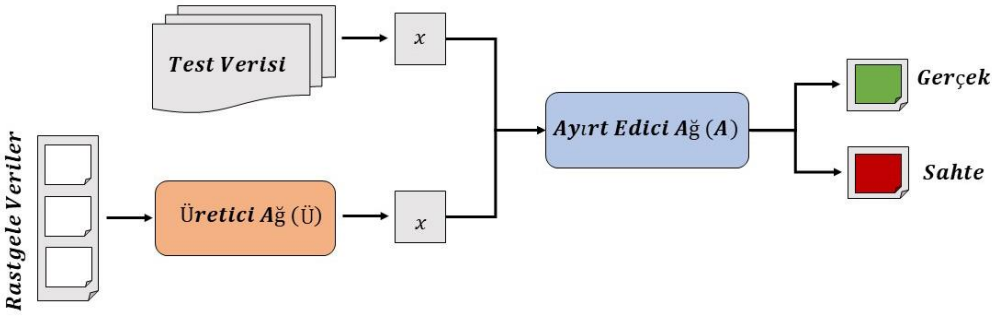


Şekil 3.2. Yapay sinir ağları grafiği.

Sinir ağları, zaman içinde doğruluklarını öğrenmek ve geliştirmek için eğitim verilerine güvenmektedir. Bununla birlikte, bu öğrenme algoritmaları doğruluk için ince ayar yapıldığında, verileri yüksek bir hızda sınıflandırmamıza ve kümelememize olanak tanımaktadır (bilgisayar bilimi ve yapay zekada güçlü araçlardır). Konuşma tanıma veya görüntü tanımadaki görevler, uzmanlar tarafından yapılan manuel tanımlamayla karşılaştırıldığında dakikalar yerine saatler sürebilmektedir. En iyi bilinen sinir ağlarından biri Google'ın arama algoritmasıdır.

3.4. Çekişmeli Üretici Ağlar (ÇÜA)

ÇÜA mimarisi ilk olarak Goodfellow ve arkadaşları tarafından 2014 yılında tanıtılmıştır (Goodfellow ve ark., 2014). Modelin mimarisi, her ikisi de derin sinir ağları olan bir Üretici Ağ (Ü) ve bir Ayırt Edici Ağ (A) olmak üzere iki ana bileşenden oluşmaktadır. Mimariye şematik bir genel bakış Şekil 3.3'te verilmiştir.



Şekil 3.3. Çekişmeli üretici ağlar mimari yapısı.

Üretici Ağlar (Ü); rastgele görüntü (z) kullanarak görüntüleri oluşturmak için kullanılan bir ağdır. Görüntü kullanılarak oluşturulan görüntüler Ü(z) olarak kaydedilmektedir. Genellikle rastgele bir nokta olan bir Gauss görüntüsü olan girdilerdir. Hem Ü hem de A ağlarının parametreleri, ÇÜA'nın eğitim süreci sırasında kendini yenileyerek güncellenmektedir.

Ayırt Edici Ağlar (A); belirli bir görüntünün gerçek bir dağılıma ait olup olmadığını belirlemek için bir ayrımcı ağ olarak kabul edilmektedir. Bir girdi görüntüsü x alır ve x'in gerçek bir dağılıma ait olma olasılığını temsil eden A (x) çıktısını üretmektedir. Çıktı 1 ise, gerçek bir görüntü dağılımını göstermektedir. Gerçek bir görüntü 0 olarak çıkış değeri alırsa, sahte bir görüntü dağılımına ait olduğunu göstermektedir.

Daha kesin olarak, X uzayında bir veri kümesi verildiğinde, üreteç girdi olarak önceki bir p_z dağılımından rastgele görüntü z alır ve bir $x = Ü(z;\theta_g) \in X$ çıktısı üretmekte, bu da θ_g üretici ağın parametreleridir. Tipik seçim önceki dağılım için p_z çok değişkenli bir Gauss dağılımıdır. Ayırıcı $A : X \rightarrow [0,1]$ ya gerçek bir veri noktası x ya da sahte bir veri noktası x ile beslenir, girdi veri noktasının gerçek olduğu ve üretici ağ tarafından oluşturulmadığı olasılığını temsil etmektedir. Ayrımcının amacı tahmin doğruluğunu en üst düzeye çıkarmak yani x'i gerçek ve x'i sahte olarak sınıflandırmaktadır. Bu aşağıdaki fonksiyonla tanımlanmaktadır.

Fonksiyon ;

$$\underset{A}{MaxV}(A, Ü) = \mathbb{E}_{x\sim p_{data}(x)}\left[log\left(d(x;\theta_d)\right)\right] + \mathbb{E}_{z\sim p_g(z)}\left[log\left(1 - A(Ü(zx;\theta_d)x;\theta_d)\right)\right] \quad (3.1)$$

Oluşturucunun amacı, gerçek olarak kabul edilen, yani yüksek A(x) değerleri olan verileri üretmektir. Bu, aşağıdaki amaç fonksiyonunda tanımlanmaktadır:

$$\underset{Ü}{MinV_Ü}(Ü) = \mathbb{E}_{z\sim p_g(z)}\left[log\left(1 - A(Ü(zx;\theta_d)x;\theta_d)\right)\right] \quad (3.2)$$

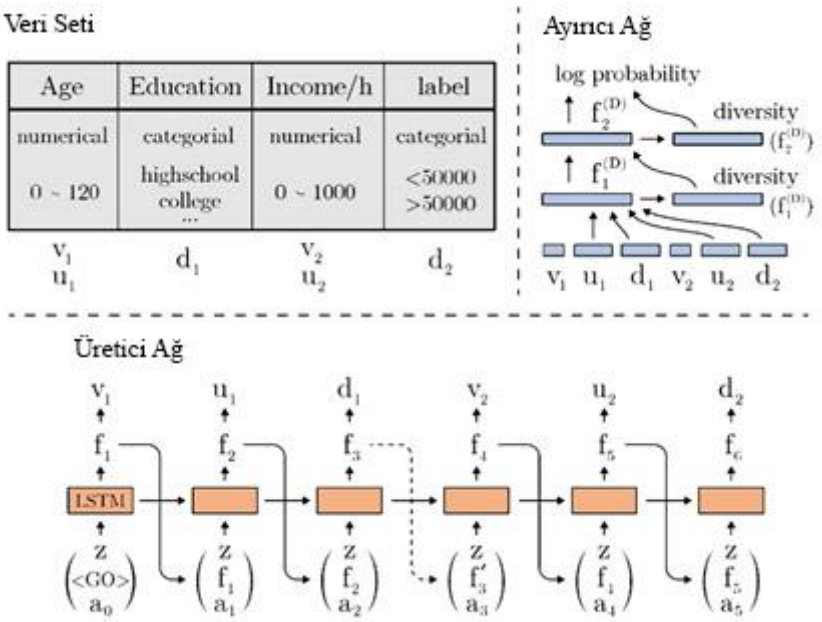
Bu iki amaç işlevi, aşağıdaki değer işleviyle iki oyunculu bir minimum-maksimum oyunda birleştirmektedir:

$$\underset{Ü}{\underset{A}{MinMaxV}}(A, Ü) = \mathbb{E}_{x\sim p_{data}(x)}\left[log\left(d(x;\theta_d)\right)\right] + \mathbb{E}_{z\sim p_g(z)}\left[log\left(1 - A(Ü(zx;\theta_d)x;\theta_d)\right)\right] \quad (3.3)$$

Ağı eğitmek için V_A ve $V_Ü$, ayırt edici ağ ve üretici ağ parametrelerini güncellemek için ayrı ayrı kullanılmaktadır. Araştırmacılar tarafından önerilen tam eğitim yöntemi, üreticinin her eğitilmesi için ayırıcıyı k kez eğitmektedir (Goodfellow ve ark., 2014). Bu, ayırt edici ağın optimum çözümüne yakın olmasını sağlamaktadır.

3.4.1. Tabular çekişmeli üretici ağlar (TGAN)

Tabular ÇÜA (TGAN), ÇÜA mimarisinin bir varyasyonudur. Lei Xu ve Kalyan Veeramachaneni tarafından 2018 yılında tanıtıldı ve herhangi bir tablo veri setinde güvenilir bir şekilde çalışacak genel amaçlı bir ÇÜA oluşturma hedefiyle tasarlanmıştır (Xu ve ark., 2018). TGAN kullanım örneği şekil 3.4'te gösterilmiştir.



Şekil 3.4. Basit bir sayım tablosu oluşturmak için TGAN'ı kullanma örneği.

TGAN, orijinal ÇÜA formülasyonunun biraz değiştirilmiş bir versiyonunu kullanılmaktadır. Ayırt edici ağın amaç fonksiyonu aynı tutularak üretici ağa en aza indirilecek fazladan bir terim eklemektedir. Fazladan terim, verilerdeki kategorik özellikler üzerinden alınan gerçek ve oluşturulan veriler arasındaki Kullback-Leibler ayrımıdır; burada $A_J^{Ü}$ ve A_J, sırasıyla oluşturulan ve gerçek verilerin kategorik özellikleridir.

$$V_{Ü} = \mathbb{E}_{z\sim p_z} \log\big(Ü\big(z;\theta_g\big);\theta_d\big) - \sum_{j=1}^{N_A} KL(A_J^{Ü} A_J), \tag{3.4}$$

Sürekli verilerde çoklu modalite sorunuyla mücadele etmek için araştırmacılar, Gauss Karışım Modellerine (GKM) dayalı moda özgü normalleştirme ön işlemeyi kullanmayı önerilmektedir (Xu ve ark., 2018). Bu yöntemleri dört adıma ayrılmaktadır (Lei ve ark., 2018).

1. Her sürekli sütun C_J için, m bileşenli bir GKM eğitilir. GKM, verilerin dağılımını, öğrenilen ortalamalar m ile Gauss dağılımının ağırlıklı toplamı olarak modeller μ_J^1,....,μ_J^m ve standart sapmalar σ_J^1 , . . ., σ_J^m olarak hesaplanmaktadır.

2. C_i'deki her $c_{i,j}$ değeri için, $\mathrm{P}_{i,j}^{1}$, . . . $\mathrm{P}_{i,j}^{m}$ olasılığını hesaplayan ve $c_{i,j}$ Gauss modlarının her birine ait $\mathrm{W}_{i,j}^{1}$, . . . $\mathrm{W}_{i,j}^{m}$ değerini almaktadır.

3. Her $c_{i,j}$ değeri için, en olası $k = argmax_l \mathrm{P}_{i,j}^{1}$ ve modunu seçilir ve buna göre normalleştirilir, $\mathrm{v}_{i,j}^{1} = \frac{c_{i,J} - \mu_J^k}{\sigma_J^k}$. $v_{i,j}$ daha sonra [−0.99, 0.99] olarak kırpılmaktadır.

4. m Etiketli kategorik bir özellik olan normalleştirilmiş $v_{i,j}$ değeri ve karşılık gelen $w_{i,j}$ modu daha sonra $c_{i,j}$ sürekli veri noktasını temsil etmek için kullanmaktadır.

TGAN uygulamasında, m sabittir ve $m = 5$'e ayarlanmıştır. Bu, modelin basit tutulmasını ve verinin $k < 5$'e mod içermesi durumunda $5 - k$ var olmayan modların öğrenilen olasılıklarının düşük olacağı ve bununla gerekçelendirilmesini sağlamaktadır. Modelin esasen sadece k kiplerini modellemiştir.

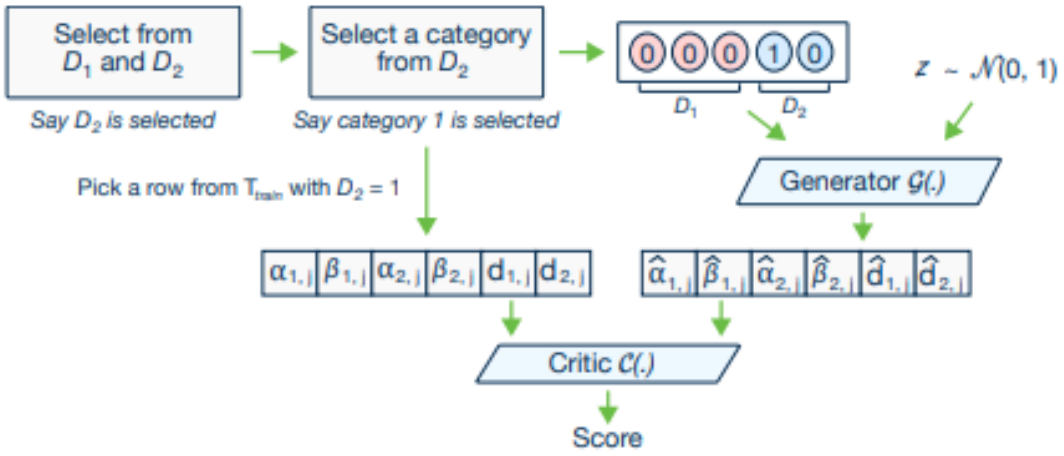
Bu moda özgü normalleştirme, üreteçte bir $tanh$ aktivasyon fonksiyonunun kullanımına izin vermek ve çok modlu sürekli verilerde mod çökmesini önlemek için önerilmiştir. Kategorik özelliklerin ön işlemesi çok daha basittir. Veriler ilk olarak one-hot olarak kodlanır ve ikinci olarak TGAN, üretici ağı kategorik özellikler için bir softmax aktivasyon fonksiyonu kullandığından, one-hot (kategorik veriler için standart yaklaşım) kodlanmış vektörlere tek tip görüntü eklemektedir. Son olarak, olasılık vektörleri olmak için normalleştirilmiştir. Hem eğitim verilerinden kategorik özelliklerin hem de mod göstergesi özelliklerinin aynı şekilde ele alınmaktadır.

TGAN'daki üretici bir Uzun Kısa Süreli Bellek (UKSB) ağıdır. Bir UKSB ağı, 1997'de Hochreiter ve arkadaşları tarafından önerilen bir tür Tekrarlayan Sinir Ağıdır (RNN) (Hochreiter ve ark., 1997). UKSB ağı, önceki girdiler hakkında yararlı bilgileri tutabilmek ve daha az önemli bilgileri unutabilmek için tasarlanmış dahili belleğe sahip bir tür sinir ağıdır. Bir ÇÜA üreteci bağlamında, belirli bir gözlemin her bir özelliğini sırayla oluşturmuştur.

3.4.2. Koşullu tablo çekişmeli üretici ağlar (CTGAN)

Koşullu Tablo ÇÜA (CTGAN), TGAN ile aynı araştırmacılar tarafından 2019'da önerilen tablo verilerini sentezlemek için tasarlanmış başka bir ÇÜA'dır (Xu, 2019; Xu

ve Veeramachaneni, 2018). TGAN'dan farklı olarak, CTGAN, Wasserstein ÇÜA kayıp fonksiyonunu, aşağıda açıklanan üretici ağ kaybında kritik bir değişiklikle, gradyan cezası ile uygulanmaktadır. TGAN'ın CTGAN'dan farklı bir yönü, sürekli verileri ön işleme yöntemidir. Sayısal özellik başına belirli sayıda m modu olan bir normal Gauss Karışım Modeli (GMM) kullanmak yerine, değişken bir Gauss Karışım Modeli uygulanmaktadır. GKM, her bir özellikte bulunan m_j modlarının miktarını tahmin etmektedir. Sayısal verilerin ön işlemesi aksi takdirde aynıdır. CTGAN, Gumbel-Softmax'ı uyguladığından, kategorik verilerin ön işlenmesinde eklenen gürültü ortadan kaldırmaktadır. CTGAN'ın genel mimarisi şekil 3.5'te gösterilmiştir.



Şekil 3.5. CTGAN modeli.

CTGAN, ÇÜA'lardaki dengesiz veri ve mod çökmesi sorununu iki yönlü bir şekilde ele almaktadır. Araştırmacılar, modelin koşullu dağılımları daha iyi öğrenmesine izin vermek için üretici ağın açık bir koşullu yapısı ve eleştirmen ile birlikte bir "örnekleme yoluyla eğitim" yöntemi önermektedir. Örnekleme yoluyla eğitim, modelin eğitimi sırasında eleştirmenin, $A_j = A_j^k$ koşulunu karşılayan üretilen ve gerçek gözlemleri karşılaştırdığından emin olmanın bir yöntemidir. Daha spesifik olarak yöntem, her bir gözlem grubu için yedi adımdan oluşmaktadır:

1. Her kategorik j sütunu için N_{A_j}uzunluğunda N_{batch} sıfır doldurulmuş "maske" vektörleri m_j oluşturmaktadır.

2. Her gözlem için, tüm kategorik sütunlardan eşit olasılıkla rastgele bir kategorik sütun A_j seçin. j_i^*, gözlemi için seçilen sütunun indeksi.

3. $A_{j_i^*}s.t.$ Değerleri üzerinde olasılık kütle fonksiyonlarını (PMF) oluşturmaktadır. Her bir değerin olasılık kütlesi, eğitim verilerindeki frekansının logaritmasına eşittir.

4. Önceki adımda pmf'den her vektör için bir k^* değeri örneklenmektedir.

5. Karşılık gelen maske vektörlerini değiştirmektedir. $s.t.\, m_{j_i^*}^{k_i^*} = 1$

6. Her gözlem için, $A_j = A_j^k$ koşulunu temsil edecek bir vektör olan $cond_i$'yi oluşturmak üzere maske vektörlerini bir araya getirmiştir.

7. Daha sonra koşul vektörleri hem eğitim verilerinden hem de üreteçten, $X_{reel} \sim P_{data}|cond$ ve $X_{synth} \sim P_A|cond$ 'dan koşulu karşılayan gözlemleri örneklemek için kullanılmıştır.

Üretici ağ koşullu örneklemeyi kolaylaştırmak için, üreteç yalnızca girdi olarak rastgele gürültüyü değil, aynı zamanda $cond$ vektörünü de alır ve üreticiyi istenen koşula eğitmek için koşul A_j^* arasındaki kaybında ekstra bir çapraz entropi terimi kullanmaktadır.

Örnekleme yoluyla eğitim ve koşullu oluşturucunun ötesinde CTGAN, karşıt ağlar olan PacGAN yapısını kullanmakta, bu da karşıt ağların aynı anda birden fazla gözlemi dikkate almasına olanak tanımaktadır (Lin ve ark., 2020). Ayrıca, bir sinir ağı aşırı uyum azaltma yöntemi olan bırakma (dropout) kullanaktadır (Srivastava ve ark., 2014).

CTGAN, üretici Ü$(z, cond)$ ağ yapısı aşağıdaki gibi tanımlanmıştır.

$$h_0 = z \oplus cond \tag{3.5}$$

$$h_1 = h_0 \oplus ReLU(BN(f^0(h_0))) \tag{3.6}$$

$$h_2 = h_1 \oplus ReLU(BN(f^1(h_1))) \tag{3.7}$$

$$A_l = Gumbel_{0.2}\left(g_l^A(h_2)\right), \quad l = 1,2,\dots,N_A \tag{3.8}$$

$$C_j = \tanh\left(g_j^C(h_2)\right), \quad j = 1,2,\dots,N_C \tag{3.9}$$

$\oplus$'nin birleştirme işlemi olduğu yerde, $f^k(x), k = 1,2$, 256 tamamen bağlı nöral katman boyutu $g_k^C(x)$, 1 boyutunda tam bağlı sinir katmanıdır ve $g_k^A(x)$'dır. N_{Ak}

Boyutunda tamamen bağlı nöral katman $(z, cond)$ öğesinin son katmana kadar üreteç aracılığıyla korunmaktadır.

Ayırıcı yapısı resmi olarak şu şekilde tanımlanmıştır.

$$h_0 = x_i \oplus \dots \oplus x_{pac} \oplus cond_i \oplus \dots \oplus cond_{pac} \quad (3.10)$$

$$h_1 = drop\left(leakyReLU_{0,2}\left(f^1(h_0)\right)\right) \quad (3.11)$$

$$h_2 = drop\left(leakyReLU_{0,2}\left(f^2(h_1)\right)\right) \quad (3.12)$$

$$C = g_C(h_2) \quad (3.13)$$

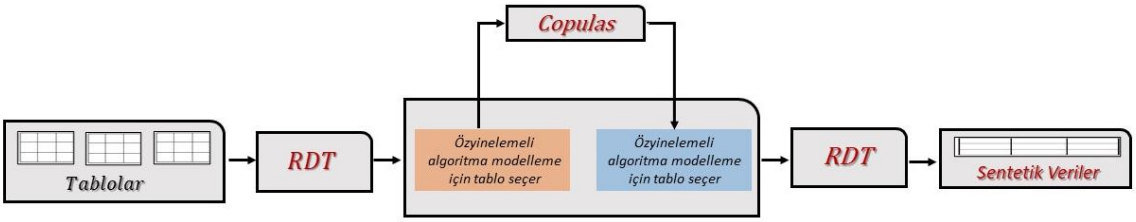
pac Bir kerede göz önünde bulundurulan gözlemlerin sayısı olduğunda, bırakma (dropout) kullanımını belirtmektedir, $f^k(x) k = 1{,}2.\ g_C(x), 256$ boyutunda tam bağlantılı bir sinir katmanıdır.

3.4.3. CopulaGAN

CopulaGAN modeli, SDV açık kaynak kitaplığında tanıtılan CTGAN'ın bir varyasyonudur (Patki ve ark., 2018). GaussianCopula aracılığıyla uygulanan Kümülatif Dağıtım İşlevi (CDF) tabanlı dönüşümden yararlanmaktadır Özellikle CopulaGAN, verileri daha kolay öğrenmek için CTGAN'ın bu alternatiflerini kullanmaktadır. Olasılık teorisine dayanarak, rastgele değişkenler arasındaki karşılıklı ilişkiyi tanımlamak için copulalar kullanılmıştır.

Copulas, rastgele değişkenler arasındaki karşılıklı ilişkiyi tanımlamak için kullanılmıştır. CopulaGAN, eğitim yöntemi sırasında veri tiplerini ve eğitim verisinin formatını öğrenmeye çalışmaktadır. Sayısal olmayan ve boş veriler, Tersinir Veri Dönüşümü (RDT) kullanılarak dönüştürülmüştür. Bu dönüşüm sayesinde, modelin her bir tablo sütununun olasılık dağılımlarını öğrenebileceği tamamen sayısal bir temsil oluşturmaktadır. Ayrıca CopulaGAN, tablonun sütunları arasındaki korelasyonu öğrenmeye çalışmaktadır. Sentetik veri kasasının yeniden tasarımından ortaya çıkan son kitaplık, SDV kitaplığının kendisidir (Montanez, 2018). Önceki iki kütüphaneden farklı olarak, SDV tamamen tek başına çalışmamaktadır. Diğer iki kütüphaneye erişim gerektirmektedir. Şekil 2.6'da açıklandığı gibi, SDV tabloları önceden işlemek için

RDT'yi kullanmakta ve daha sonra bunları modelleme için yinelemeli olarak Copulas'a aktarmaktadır. Bu işlem kök tablo modellenene kadar devam etmektedir. Şekil 3.6'da SVD akış diyagramı gösterilmektedir.



Şekil 3.6. SDV akış diyagramı.

Copulas**,** tek değişkenli copulalar ve çok değişkenli copulalar olmak üzere ikiye ayrılmaktadır.

Tek Değişkenli Copulalar; Tek değişkenli olasılık dağılımlarının birçok türü vardır. Örneğin, kopulas kitaplığı hem normal hem de tek biçimli dağılımları desteklemektedir. Verilerimizde karşılaşılabilecek dağılımların çeşitliliğini hesaba katmak için, copulas kitaplığı bir temel tek değişkenli dağıtım sınıfına sahiptir. Temel sınıfın amacı, tüm alt sınıfların aynı yöntemlere sahip olmasını sağlamaktır. Bir kopula oluşturmak için, çok değişkenli dağılımın tek değişkenli dağılımlarının her birinden belirli işlevsellik elde edilmelidir (Montanez, 2018). Bu nedenle temel sınıf, aşağıdaki yöntemleri uygulamıştır:

- Uygun
- PDF
- CDF
- TersCDF
- Örneklem

Her belirli dağıtım türü bu temel sınıftan miras olarak alınmaktadır. Öznitelikler dağıtıma bağlı olarak değişecektir, ancak yöntemler her zaman aynı kalacaktır. Bunun nedeni, farklı dağılımların farklı özellikleri takip etmesi, ancak hepsinin bir olasılık yoğunluk fonksiyonuna (PDF) ve kümülatif dağılım fonksiyonuna (CDF) sahip olmasıdır. Fit yöntemi, tamamen sayısal bir diziyi alır ve dağıtım için gerekli özniteliklerini çıkarmaktadır. Örneğin normal dağılımı alırsak, takip ettiği iki öznitelik

verinin ortalaması ve standart sapmasıdır (Aas,, 2009). Bu öznitelikler kullanılarak bir PDF ve CDF hesaplanabilmektedir.

Çok Değişkenli Copulalar; Kütüphanenin ikinci bölümü, kopulaların kendilerini temsil etmekten sorumludur. Bir kopulayı girdi olarak aldığı rastgele değişkenlere bazı değişiklikler uygulayan çok değişkenli bir dağılım olarak düşünülmektedir. Bir copula temsilinin tek değişkenli dağılımlardan çok farklı olmadığını görülmektedir. Aslında, genel yapı neredeyse aynıdır. Unutulmaması gereken ilk şey, bazı copula ailelerinin yalnızca iki değişkenli , diğerlerinin ise çok değişkenli olduğudur (Montanez, 2018). Ayrıca, çok değişkenli kopulaların bazıları, uygulamalarında iki değişkenli kopulaları kullanmaktadır. Bu nedenle, copulas kitaplığının merkezinde, biri iki değişkenli modellere diğeri çok değişkenli modellere yönelik olmak üzere iki temel sınıfa ayrılmıştır. Her ikisi de benzer işlevleri desteklemektedir. Temel fark, çok değişkenli sınıfın, uygulamasında hangi iki değişkenli sınıfın kullanılacağını belirleme seçeneği sunmasıdır.

Temel tek değişkenli dağıtım sınıfına benzer şekilde, temel kopula sınıfları aşağıdaki yöntemleri uygulanmıştır:

- Uygun
- PDF
- CDF
- Örneklem

Buradaki yöntemler ile tek değişkenli dağılım için olan yöntemler arasında birkaç fark vardır. Genel olarak, bir CDF işlevi girdi olarak x değerini alır ve rastgele değişkenin x'e eşit veya daha küçük bir değer almasını sağlamaktadır. Birden çok boyutta, CDF işlevi birden çok girdi alır ve tek bir olasılık döndürmektedir.

Her bir özel copula türü, bu temel sınıflardan birinden miras almaktadır (Barthelme, 2011). Örnek olarak, SDV tarafından kullanılan varsayılan copula olduğu için Gauss Copula'ya incelenmiştir. Aşağıdaki adımlar kullanılarak bir Gauss Copula oluşturulmuştur.

1. İlk önce $X_0, X_1, \ldots, X_n$ tablosundaki sütunları çağırarak başlanmıştır, (Burada n sütun sayısıdır).
2. Her X_i, sütunu için tek değişkenli bir dağılım örneği oluşturulmuştur.

3. Bu dağılımlardan $F_0, F_1, \dots, F_n$ CDF'lerini elde edilmektedir.

4. Tablodaki her satır için değerleri aşağıdaki dönüşümü kullanılmıştır:

$\emptyset^{-1}(F_i(x_i))$, burada $\emptyset^{-1}$ standart bir normal dağılımın ters CDF'sidir ve F_i, aşağıdaki dönüşümün CDF'sidir ve x_i için elde edilen dağılımdır.

5. Bu, copula fonsiyonunu vermektedir,

$$Y = \emptyset^{-1}\left(F_0(x_0)\right) + \emptyset^{-1}\left(F_1(x_1)\right) + \cdots + \emptyset^{-1}(F_n(x_n)). \quad (3.14)$$

6. Son olarak, bu dönüştürülmüş değerler için kovaryans matrisi Σ hesaplanmıştır.

Bu adımların tümü, sınıfın fit yönteminde tamamlanmaktadır. fit işlemi sırasında oluşacak hatalardan kaçınmak için, fit yöntemi, giriş tablosunun tamamen sayısal olması gerektirmektedir. Bu yapıldıktan sonra, veriler bu yeni dağıtımdan örnekler üretilebilmektedir. Örnek yöntemi çağrıldığında, veriler ilk olarak çok değişkenli Gauss dağılımı kullanılarak oluşturmakatadır. Daha sonra veriler, her bir sütuna $F_i^{-1}(\emptyset(X_i))$. dönüşüm uygulanarak orijinal dağılıma uyan bir forma geri döndürülmektedir Bu durumda, F_i^{-1}, orijinal tek değişkenli dağılımın ters CDF'sidir ve Ø, standart bir normal dağılımın CDF'sidir. Uyum ve örnekleme yöntemlerinin uygulanması, her bir copula için oldukça farklıdır. Bu nedenle temel çok değişkenli kopula sınıfı bulunmaktadır. Herhangi bir katkıda bulunan kişi, bu yöntemler var olduğu sürece kitaplığa yeni bir copula türü ekleyebilmektedir.

3.5. Değerlendirme Metrikleri

Bir ÇÜA modelini değerlendirmek, çeşitli metrikler farklı sonuçlara yol açabileceğinden, basit bir yöntem değildir. Spesifik olarak, bir değerlendirme metriğindeki iyi bir performans, başka bir metrikte iyi performans garanti etmemektedir (Theis, 2015).

Sentetik veriler, tablo halinde veri oluşturma görevi için uygun olan yeterli sayıda metriğe karşı değerlendirmektedir. Değerlendirme, gerçek veri $Treal$ tablosunun yanı sıra eğitilmiş G üreticisinden, oluşturulan sentetik veri $Tsyn$ tablosunda

gerçekleştirmektedir. Değerlendirme yöntemleri Görsel, İstatistiksel ve Makine Öğrenimi tabanlı olmak üzere üç alt kategoriye ayrılmaktadır.

3.5.1. Görsel değerlendirme

Üretilen verilerin görsel temsili, Ü'nün gerçek verilerin özelliklerini muhafaza edip edemediğini analiz ederek Üreteci Ağının (Ü) performansını değerlendirmek için güçlü bir yöntemdir. Buna dayanarak, insanlar sonuçları kolayca doğrulayabilir ve gerçek ve sentetik veriler arasındaki benzer kalıpları tanıyabilmektedir. Görsel değerlendirme, Kümülatif Toplamlar ve Sütun Korelasyonuna dayalı olarak gerçekleşmektedir.

Gerçek ve sentetik veriler için her sütunun dağılım grafiği, herhangi bir gizli ilişkiyi ortaya çıkarmasa da hızlı ve mantıklı bir kontrol sağlamaktadır. Bu yöntem, üretilen ve gerçek verilerin istatistiksel özelliklerinin birbirine benzer olup olmadığını gösterebilmektedir.

Gerçek ve sentetik veriler için her sütunun Kümülatif Toplamı, sütun başına dağılımlar arasındaki benzerliği göstermek için görselleştirilmektedir. Bu görselleştirme hem kategorik hem de sürekli sütunlar için yararlı bir anlayış sunmaktadır. Ancak bu yöntem, sütunlar arasındaki ilişkiler hakkında herhangi bir fikir sağlayamamaktadır.

Başka bir değerlendirme yöntemi, tablonun her bir sütunu arasındaki ilişkiyi gösteren Korelasyon tablosuna dayandırılmaktadır. Gerçek ve sentetik verilerin korelasyon matrisinin karşılaştırılması, oluşturucunun tablonun sütunları arasındaki ilişkiyi uygun şekilde modellemeyi başarıp başarmadığını gösterebilmektedir.

3.5.2. İstatistiksel değerlendirme

Gerçek ve oluşturulmuş tablolara farklı türde istatistiksel testler uygulanabilmektedir. Özellikle bu metrikler, gerçek tablonun tek tek sütunlarını, oluşturulan verilerden karşılık gelen sütunla karşılaştırır ve bu analizin bir sonucunu üretmektedir. Deprem veri seti ile eğitilmiş ÇÜA modellerinin değerlendirilmesi için iki istatistiksel test kullanılarak gerçekleştirilmektedir. Bunlar, KSTest ve CSTest'tir.

KSTest, iki örnekli Kolmogorov-Smirnov testi ve ampirik Kümülatif Dağıtılmış Fonksiyonu (CDF) kullanarak gerçek ve oluşturulan tablo arasındaki sürekli öznitelikler

için dağılımları karşılaştırmaktadır. Her sütun karşılaştırmasının sonucu, 1 eksi $Test\ D$'dir Gözlemlenen ve beklenen CDF değerleri arasındaki maksimum mesafeyi ifade etmektedir (Patki ve ark., 2018).

Ayrık değerler için, *CSTest* metriği, gerçek ve sentezlenmiş veriler arasındaki sütun dağılımlarını karşılaştırmak için $Ki-kare$ testinden yararlanmaktadır (Patki ve ark., 2018). Karşılaştırma sütun sütun gerçekleştirmekte ve sonuç $CSTest\ p$ değeridir. Yani iki sütundaki değerlerin aynı dağılımdan örneklenmiş olma olasılığıdır.

3.5.3. Makine öğrenimi tabanlı değerlendirme

Bu ölçüm ailesi, oluşturulan verilerin kalitesini değerlendirmek için Makine Öğrenimi algoritmalarından yararlanmaktadır. $Treal$ ve $Tsyn$'in sahip olduğu ilişkiler hakkında içgörü bilgisi sağlayabilmektedir. Özellikle, tablo halinde veri oluşturma görevi için ÇÜA'lar, makine öğrenimi verimlilik ölçümlerinin yanı sıra algılama ölçümleriyle değerlendirilmmektedir.

Algılama Metrikleri; oluşturulan verileri gerçek verilerden ayırmanın ne kadar zor olduğunu değerlendirmektedir. Yöntem olarak; bu metrikler, girdi verilerinin sentetik mi yoksa gerçek mi olduğunu tahmin eden makine öğrenimi modellerine dayanmaktadır (Patki ve ark., 2018). Bu nedenle, her veri kaydına, gerçek mi yoksa oluşturulmuş mu olduğunu gösteren bir bayrak ilişkilendirilmiştir. Daha sonra, bayraklı veriler karıştırılarak ve bayrak tahmin edilmeye çalışılarak makine öğrenimi modelleri çapraz doğrulanmaktadır. Son olarak, bu metriklerin sonucu, 1 eksi tüm çapraz doğrulama bölmelerinin ortalama performans ölçümü puanına eşitlenmektedir (Patki ve ark., 2018). Bu metriklerde kullanılabilecek makine öğrenimi modelleri Lojistik Regresyon veya SVD sınıflandırıcılarıdır.

Makine Öğrenimi Verimlilik Metrikleri; Makine Öğrenimi modellerini kullanarak sorunları çözmek için gerçek verileri oluşturulan verilerle değiştirmenin mümkün olup olmadığını göstermektedir. Özellikle bir model *Tsyn* üzerinde eğitilir ve ardından *Treal* üzerinde test edilmektedir (Patki ve ark., 2018). Sınıflandırma sorunları olması durumunda Karar Ağacı, AdaBoost veya MLP sınıflandırıcı kullanılmaktadır. Bu modellerin performansı doğruluk ve F1 puanına göre değerlendirilmektedir. Regresyon görevleri için, makine öğrenimi modelleri olarak Lineer Regresyon veya MLP(Çok Katmanlı Algılayıcı) regresyon kullanılmaktadır (Patki ve ark., 2018). Farklı modellerin

ortalama performansı, üretici ağın değerlendirilmesi için metrik olarak kullanılabilmektedir.

3.6. Materyal

3.6.1. Veri seti

Bu çalışmada, veri seti İstanbul Büyükşehir Belediyesi açık veri portalı tabanından alınmıştır. 2019-2020 tarih aralığında İstanbul çevresindeki gerçekleşen depremlerin verilerini içermektedir (İBB Açık Veri Portal, 2021). Tablo verilerimiz 349 satır ve 8 sütundan oluşmaktadır. Deprem veri seti çizelge 3. 1'de gösterilmiştir.

Çizelge 3.1. İstanbul ve çevresindeki oluşan depremlerin veri seti.

	EventID	zaman	enlem	boylam	derinlik/km	büyüklük türü	büyüklük	konum adı
0	20200204_0000090	2020-02-04T10:17:09.4Z	38.98	27.86	6	ml	3.4	WESTERN TURKEY
1	20200204_0000082	2020-02-04T08:43:16.0Z	38.98	27.87	10	ml	4.1	WESTERN TURKEY
2	20200204_0000067	2020-02-04T06:13:29.2Z	38.42	25.51	15	ml	3.3	AEGEAN SEA
3	20200204_0000046	2020-02-04T03:42:21.5Z	38.40	25.53	17	ml	3.3	AEGEAN SEA
4	20200204_0000034	2020-02-04T03:14:45.4Z	39.08	27.83	9	ml	3.3	WESTERN TURKEY
...	...	...	...	...	...	...	...	...
344	20190429_0000132	2019-04-29T18:39:50.8Z	39.42	26.36	7	ml	3.8	NEAR THE COAST OF WESTERN TURKEY
345	20190429_0000124	2019-04-29T18:02:43.3Z	39.38	26.29	10	ml	4.4	NEAR THE COAST OF WESTERN TURKEY
346	20190428_0000079	2019-04-28T14:49:25.1Z	38.69	26.90	5	ml	3.4	NEAR THE COAST OF WESTERN TURKEY
347	20190412_0000074	2019-04-12T13:03:47.8Z	40.69	30.61	7	ml	4.0	WESTERN TURKEY
348	20190402_0000092	2019-04-02T16:07:43.6Z	38.59	31.14	5	ml	3.2	WESTERN TURKEY

Türkiye'nin en büyük şehri olan İstanbul ve çevresinde meydana gelen depremlerin analizi yapılarak sentetik veriler oluşturulmaktadır. Böylece gelecekte oluşabilecek depremlerle ilgili araştırmalarda elde edilen sentetik verilerden yararlanılarak, olası bir afet durumunda önceden hazırlıklı olunması amaçlanmaktadır.

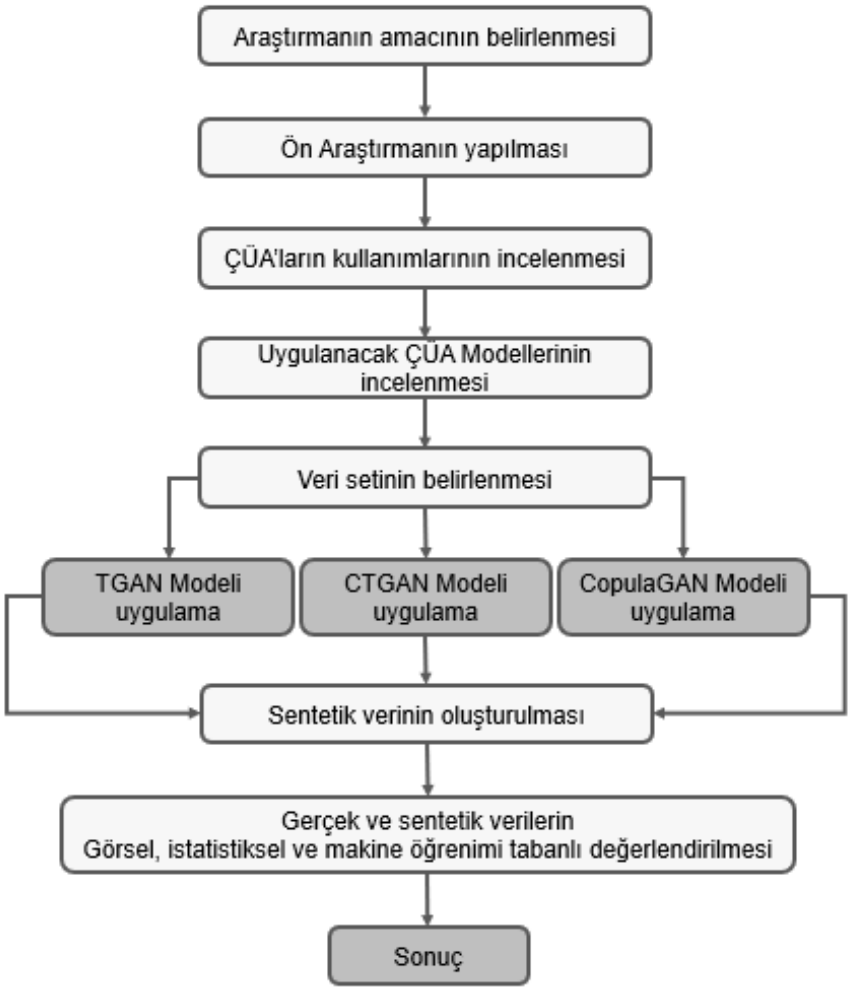
3.6.2. Kullanılan programlar

Bu çalışmaya ait uygulama için Google Colaboratory, JupyterLab ve Microsoft Excel programları kullanılmıştır. Colab, herkesin tarayıcı aracılığıyla rastgele python kodu yazmasına ve yürütmesine izin veren ve özellikle makine öğrenimi, veri analizi ve eğitim için çok uygun bir yazılımdır. Daha teknik olarak, Colab, GPU'lar dahil bilgi işlem

kaynaklarına ücretsiz erişim sağlayan, kullanmak için kurulum gerektirmeyen bir Jupyter dizüstü bilgisayar hizmetidir. JupyterLab, not defterleri, kod ve veriler için en yeni web tabanlı etkileşimli geliştirme ortamıdır. Esnek arayüzü, kullanıcıların veri bilimi, bilimsel bilgi işlem, hesaplamalı gazetecilik ve makine öğrenimindeki iş akışlarını yapılandırmasına ve düzenlemesine olanak tanımaktadır.

3.7. Yöntem

ÇÜA ile sentetik veri üretimi yapmak için birçok farklı yöntem kullanılabilmektedir. Bu yöntemlerin uygulanan problem ve istenilen sonuca göre başarı oranları değişmektedir. Bu çalışmada sentetik veri üretimi ve üretilen sentetik verinin gerçek veriye yakınlığını ölçmek için uygulanan yöntemin akış diyagramı Şekil 3.7'de gösterilmektedir



Şekil 3.7. Çalışmada izlenen yöntem.

Gerçek veri seti 349 satır ve 8 sütundan oluşmaktadır. Gerçek veri setinde sonuca etkisi olamayan işlem süresini artıracak olan (EventID, büyüklük türü ve komum adı) sabit sütunlar veri setinden çıkarılmıştır. Belirtilen etiketleri satırlardan veya sütunlardan çıkarmak için $Drop()$ Parametresi kullanılmıştır. Satırda boş parametreleri olup olmadığını kontrol etmek için $dropna()$ parametresi kullanarak veri setinin tüm eksik değerleri (NaN değerler) kaldırılmıştır. Veri setinin sütunlarının ve veri tiplerinin neler olduğunu öğrenmek için $Shape()$ ve $dtype()$ parametresi kullanılmıştır. Tanımlanmayan veri tiplerinin $astype()$ parametresi kullanarak veri dönüşümleri yapılmıştır. Sonuç olarak, temizleme işlemleri sonunda yeni veri setimiz 349 satır ve 5 sütundan oluşmaktadır.

TGAN, CTGAN ve CopulaGAN modellerini kullanarak sentetik veri üretmek için; TGAN, $tgan.model.TGANModel$ sınıfını içe aktarmamız ve onu $continuous\_columns$ benzersiz argümanla çağırmamız gerekmektedir. CTGAN ve CopulaGAN ise $sdv.tabular.CTGAN$ ve $sdv.tabular.CopulaGAN$ sınıfını içe aktarmamız gerekmektedir. Bu durumlar varsayılan parametrelerle TGAN, CTGAN ve CopulaGAN örneği oluşturacaktır. TGAN, CTGAN ve CopulaGAN örneğimiz olduğunda, yerleştirme işlemini başlatmak için daha önce yüklediğiniz gerçek veri setini sığdırma (fit) yöntemini kullanarak model yerleştiriyoruz. Model yerleştirildikten sonra, istenen miktarda verinin $sample$ yöntemini kullanarak aynı format ve istatistiksel özelliklere sahip sentetik veriler oluşturulmaktadır.

Çekişmeli Üretici Ağ modellerinden oluşan sentetik veri setleri, bu çalışmanın bölüm 3.5'te açıklanan metrikler kullanılarak değerlendirilmektedir. Dağıtım ve kümülatif toplam grafikleri Table Evaluator kitaplığı kullanılarak oluşturulurken, istatistiksel ve makine öğrenimi tabanlı metrikler SDV kitaplığının tek tablo metriklerine dayalı olarak hesaplanmaktadır.

4. BULGULAR

4.1. Temel İstatistikler

Bu çalışmada, üç ÇÜA modeli (TGAN, CTGAN ve CopulaGAN) ile halka açık bir veri seti ile değerlendirilmiştir. İlk değerlendirmeden başlayarak, ÇÜA modellerinin eğitim süreleri hesaplanmıştır. Modellerin eğitim süreleri çizelge 4.1'de gösterilmektedir.

Çizelge 4.1. Model eğitim süreleri.

Model	Eğitim Süreleri (dk.)
TGAN	11.28
CTGAN	00.12
CopulaGAN	00.15

Çizelge 4.1'de modeller sentetik veri oluştururken geçen süreleri hesaplamaktadır. Süreler sistem tarafından kaydedilmektedir. CTGAN ve CopulaGan modelleri daha kısa sürelerde modellerken, TGAN uzun sürede modellemeyi gerçekleştirmektedir. TGAN uzun sürede modellemeyi gerçekleştirmesinin temel nedeni, veri kümesinin parti sayısıdır (Batch size). TGAN modeli, veri setinin fazla büyük olduğundan, birkaç küçük gruba bölerek işlem yapmaktadır.

Çizelge 4.2, 4.3, 4.4 ve 4.5'de deprem verilerinin gerçek ve model eğitimlerinden sonra üretilen sentetik verilerin örnekleri gösterilmektedir. Aşağıdaki verilerde gerçek ve modeller tarafından üretilen sentetik verilerin ilk 5 satırları gösterilmektedir.

Çizelge 4.2. Gerçek veriler.

	zaman	enlem	boylam	derinlik/km	büyüklük
0	2020-02-04T10:17:09.4Z	38.98	27.86	6	3.4
1	2020-02-04T08:43:16.0Z	38.98	27.87	10	4.1
2	2020-02-04T06:13:29.2Z	38.42	25.51	15	3.3
3	2020-02-04T03:42:21.5Z	38.40	25.53	17	3.3
4	2020-02-04T03:14:45.4Z	39.08	27.83	9	3.3

Çizelge 4.3. TGAN tarafından üretilen sentetik veriler.

	zaman	enlem	boylam	derinlik/km	büyüklük
0	2020-01-23T13:09:17.6Z	39.095997	27.540858	8	3.319240
1	2020-01-27T23:46:58.6Z	38.976592	27.540858	8	3.319247
2	2020-01-25T11:55:42.7Z	39.029127	26.257380	13	2.941772
3	2020-01-28T14:08:50.7Z	39.608374	27.541862	7	2.943073
4	2020-01-31T08:23:33.4Z	39.001741	27.540858	7	2.941978

Çizelge 4.4. CTGAN tarafından üretilen sentetik veriler.

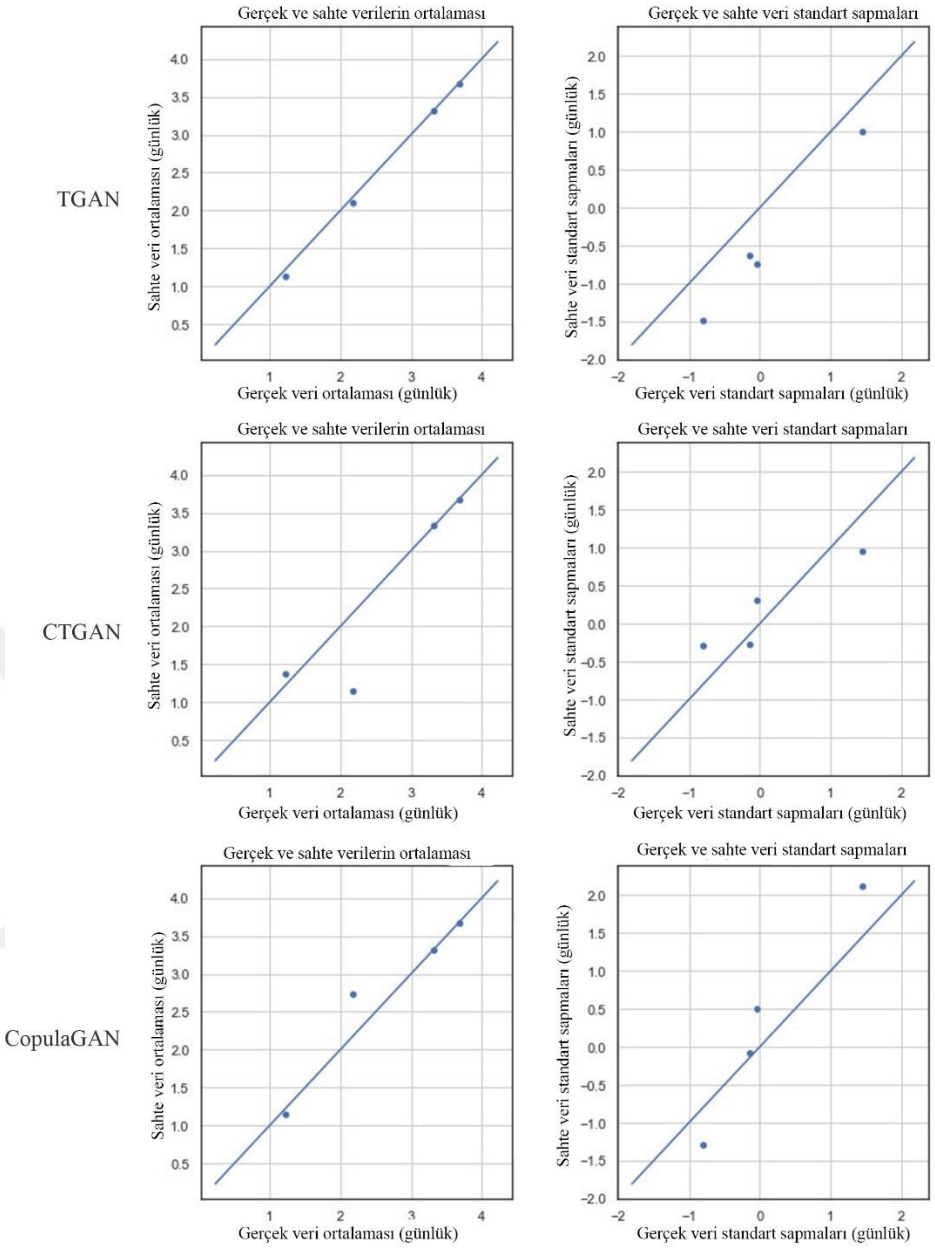
	zaman	enlem	boylam	derinlik/km	büyüklük
0	2020-02-04T01:30:09.6Z	40.82	28.10	7	3.1
1	2019-05-11T06:46:02.2Z	39.07	27.94	1	3.0
2	2019-08-08T11:02:29.7Z	39.82	29.28	5	3.3
3	2020-01-23T05:34:54.9Z	39.10	28.40	1	3.1
4	2019-04-30T18:22:35.7Z	39.15	27.52	3	3.9

Çizelge 4.5. CopulaGAN tarafından üretilen sentetik veriler.

	zaman	enlem	boylam	derinlik/km	büyüklük
0	2019-12-30T14:32:49.7Z	39.64	27.84	12	3.4
1	2019-09-26T12:17:09.6Z	39.01	27.42	22	3.8
2	2020-02-04T03:14:45.4Z	39.46	27.91	40	3.0
3	2020-01-22T21:11:23.6Z	40.64	27.84	25	3.3
4	2019-10-09T00:48:12.2Z	38.77	27.89	11	3.0

4.2. Değerlendirme Metrikleri

Sütun bazında ortalama ve standart sapma, gelişmiş bir ölçüm değildir ve herhangi bir gizli ilişkiyi ortaya çıkarmaz, ancak hızlı ve mantıklı bir kontrol işlevi görmektedir. Her sütunun ortalamaları ve standart sapmaları günlük ölçekte çizilmektedir. Çizilen değerler köşegeni (x=y) takip ederse, verilerin ortalamaları ve standart sapmaları karşılaştırılabilir olduğunu göstermektedir. Şekil 4.1'de Gerçek ve sahte veri setinin ortalama ve standart sapmaları gösterilmektedir.



Şekil 4.1. Gerçek ve sentetik veri setinin ortalama ve standart sapmaları (x=y).

Şekil 4.1'de modellerin bu özellikleri göreceli kolaylıkla yakaladığı gözlemlenmektedir. TGAN ve CTGAN, bu değerleri üretmekte iyi olduklarını, ancak CopulaGAN'nın bu değerleri üretirken zorlandığı gözlemlenmiştir. Standart Sapmaları bütün modellerimizin yakalamada zorlandığı gözlemlenmiştir. Çoğu model ortalamaları yakalayabilir, ancak standart sapmaları yakalamakta oldukça zorlanabilmektedir. Sentetik veriler, gerçek verilerden büyüklük sırasına göre farklılık gösteriyor gibi görülmektedir ancak standart sapmaları aynı büyüklük düzeninde tutmaktadır. Bu veri

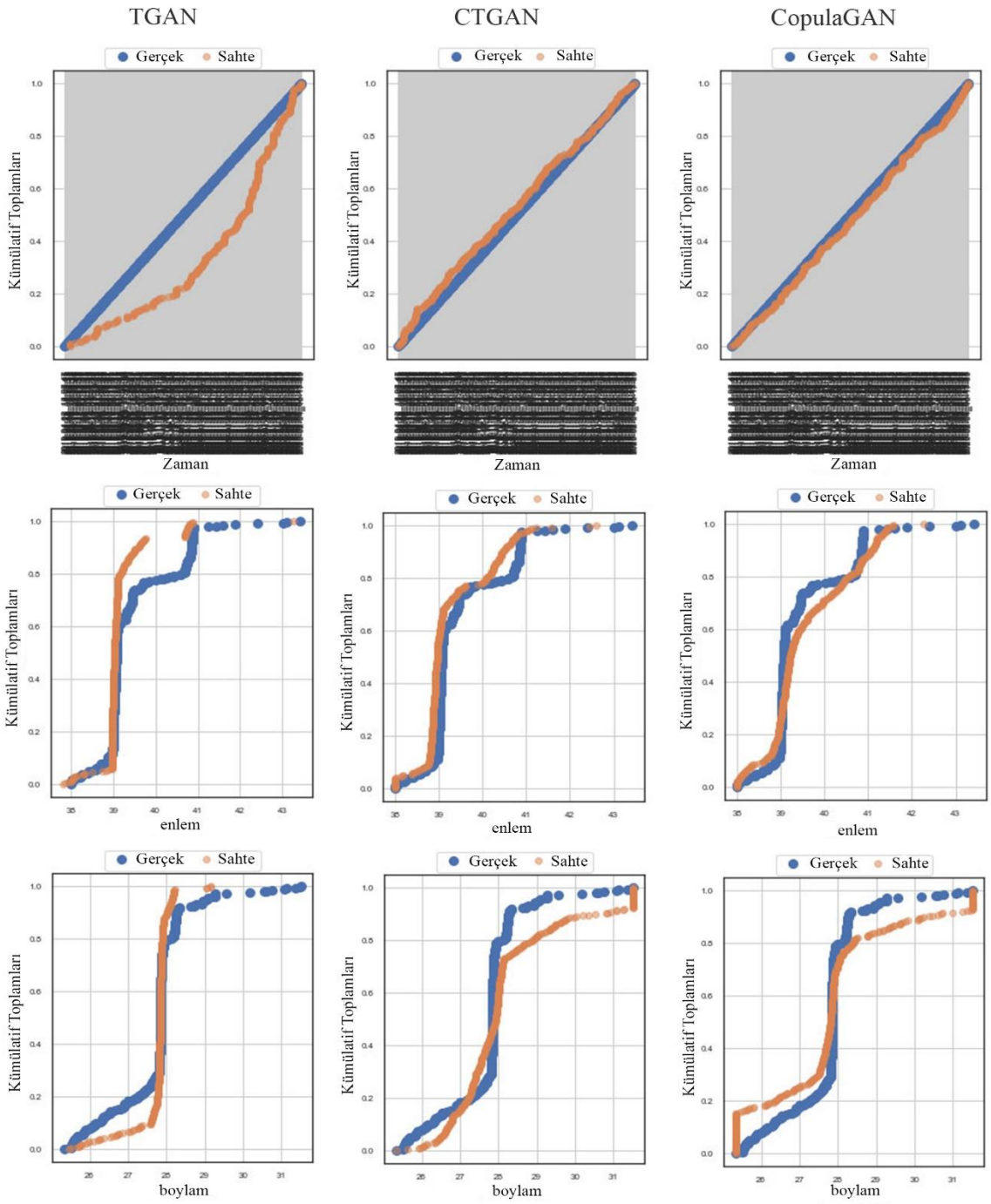
kümesinin sentezlenmesinin çok daha zor olduğunu göstermiştir. Çizelge 4.6'da gerçek ve sentetik verilerin ortalamaları ve standart sapmalarını göstermektedir.

Çizelge 4.6. Veri setinin ortalama ve standart sapmaları.

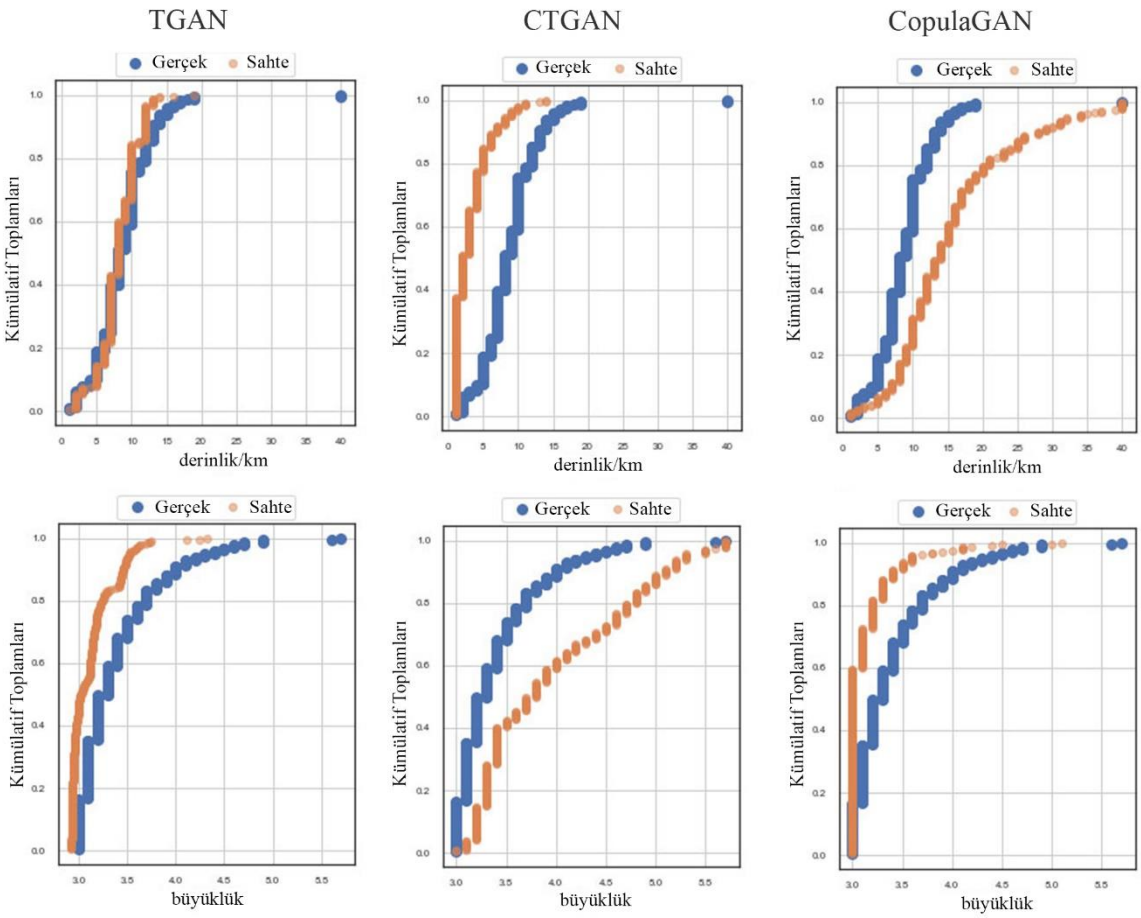
	Gerçek Veriler	TGAN	CTGAN	CopulaGAN
Ortalama	19.8443	19.5339	21.3715	21.4546
Standart Sapma	1.63223	0.9953	1.8125	2.8049

Çizelge 4.6'da TGAN ile üretilen sentetik veriler, gerçek veri setine yakın ortalamaya sahiptir. CTGAN ve CopulaGAN ise bu değerleri yeniden üretmekte zorlandığı gözlemlenmektedir. Standart Sapmayı ise en iyi yakalayan CTGAN'dır. Sonrasında TGAN ve CopulaGAN gelmektedir.

Grafikler, bir ÇÜA'nın belirli özelliklerde yanlış dağılımları oluşturup oluşturmadığını manuel olarak görmenin en iyi yoludur. Sütun başına dağılımlar arasındaki benzerliği görsel olarak incelemek, hem gerçek hem de sentetik veriler için her sütunun kümülatif toplamları şekil 4.2 ve şekil 4.3'de çizilmektedir. Bu, yalnızca grafik ile bir sütun hakkında oldukça kapsamlı bir anlayış sağlar ve hem kategorik hem de sürekli sütunlar için çalışmaktadır. Bu grafiğin sütunlar arasındaki ilişkiler hakkında herhangi bir fikir vermediğini ve tüm tablo için sınırlı temsil gücü sağladığını unutmamak gerekir. Hangi tür değerlerin ve sütunların diğerlerinden daha kolay veya daha zor olduğunu belirlemeye izin vermektedir.



Şekil 4.2. Gerçek ve sentetik veri kümesinin kümülatif toplamları (1).



Şekil 4.3. Gerçek ve sentetik veri kümesinin kümülatif toplamları (2).

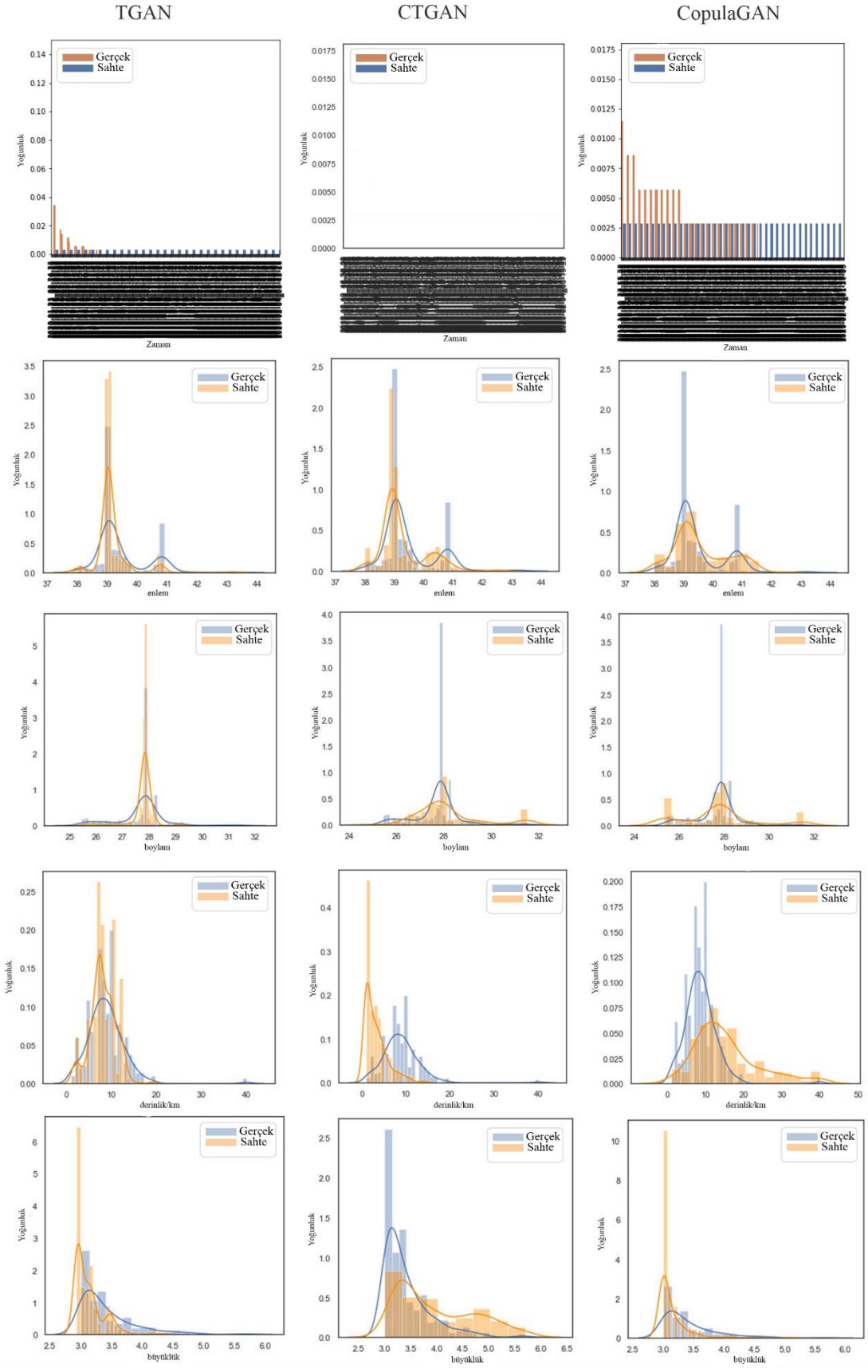
Şekil 4.2 ve şekil 4.3'de Mavi, gerçek verileri, turuncu ise sentetik verileri göstermektedir. Modeller tarafından oluşturulan gerçek ve sentetik veri kümesinin yedi sütununun kümülatif toplamları gösterilmektedir. TGAN ortalamadan daha uzak değerleri yakalamada, dağılımın kuyruklarına ulaşmada daha iyi olduğu ortaya çıkmaktadır. Ancak CTGAN ve CopulaGAN dağılımın kuyruklarına ulaşmada zorluk yaşadığı gözlenmiştir. Veri kümesinin kümülatif toplamları çizelge 4.7'de göstrilmektedir.

Çizelge 4.7. Veri kümesinin kümülatif toplam sayıları.

	Kümülatif Toplamları
Gerçek Veriler	27702.58
TGAN	27269.30
CTGAN	29834.55
CopulaGAN	29950.67

Çizelge 4.7’de modeller tarafından üretlen sentetik verilerin kümülatif toplamları, gerçek veri setine en yakın kümülatif toplamları veren TGAN modelidir. Sonra CTGAN ve CopulaGAN modelleri takip etmektedir.

Şekil 4.4‘de, ÇÜA modellerinin temel alınan verilerin istatistiksel özelliklerini ne kadar iyi koruduğunu göstermektir. Bunun için, kutu grafiklerini ve gerçek-sentetik veri kümesinin sütün özellikleri arasındaki ikili ilişkileri karşılaştırılmaktadır. Ek olarak, sentetik veri seti ile gerçek veri seti kümesiyle karşılaştırmak için özellik başına dağılım grafiklerinin görsel incelemesi kullanılmaktadır. Grafikler, gerçek veri sütunlarına benzeyen örnekleri elde etmek için ters dönüşüm fonksiyonu kullanmaktadır.



Şekil 4.4. Gerçek ve sentetik veri kümesinin özellik başına dağılım grafiği.

Şekil 4.4 ‘de Mavi, gerçek verileri, turuncu ise sentetik verileri göstermektedir. Özellik başına dağılımın görsel bir incelemesinden, TGAN, en iyi performans gösterdiği gözlenmiştir. CTGAN ve CopulaGAN ise bazı sütünlarda üretmiş olduğu verilerde mevcut olmayan çok modlu model ile görünen mod çökmesinin bir kombinasyonundan zorlandığından dolayı en kötü performansı göstermiştir

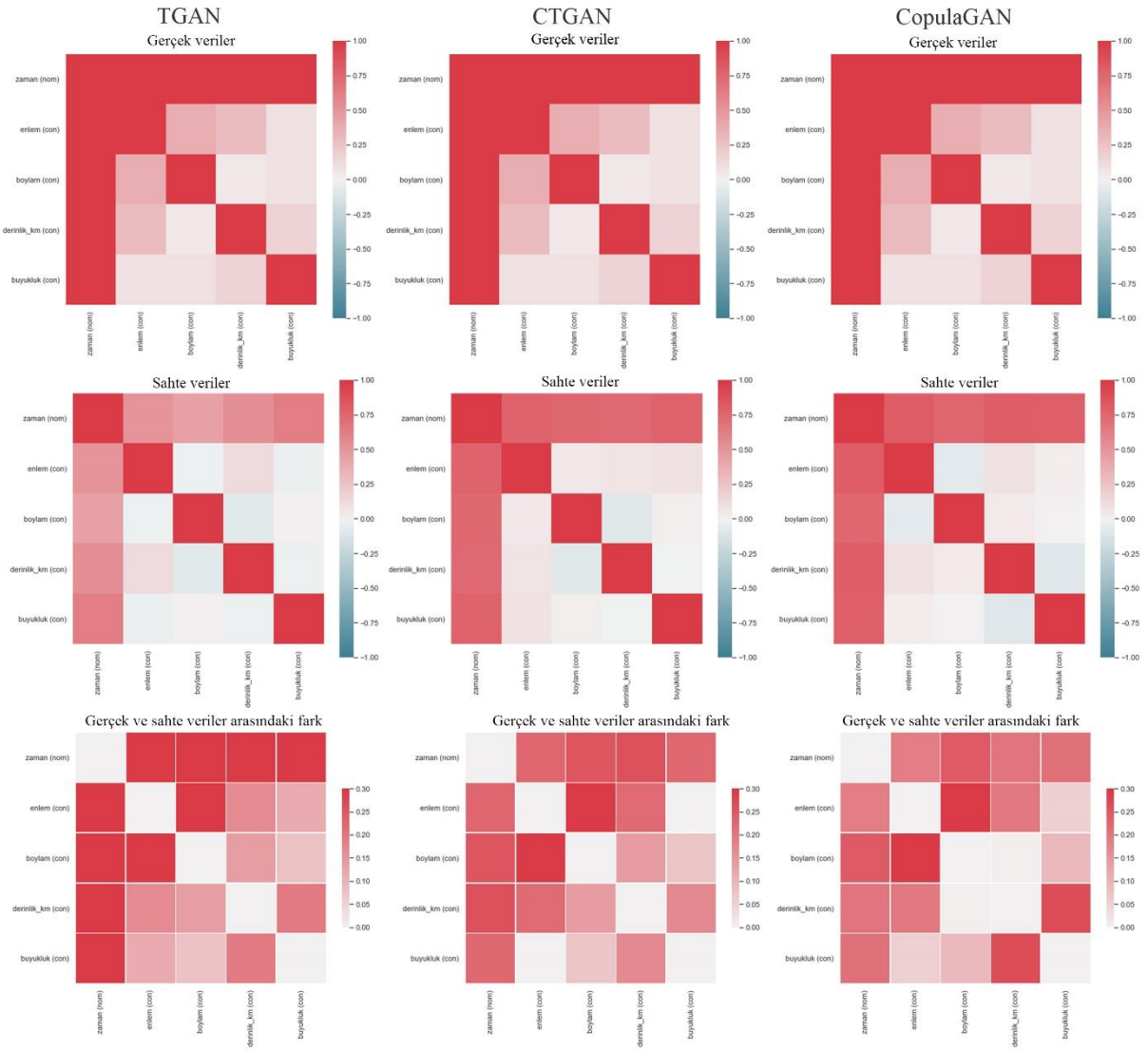
İstatistiksel ve algılama metrikleri, üzerlerinde farklı türde istatistiksel testler çalıştırarak tabloları karşılaştırmaktadır. Bu metrikler gerçek tablodaki sütun değerleri, sentetik tablodaki karşılık gelen sütundaki değerleri ile karşılaştırır ve sonunda testten elde edilen ortalama sonucu vermektedir. Çizelge 4.8’de istatistik ve algılama metrikleri gösterilmektedir.

Çizelge 4.8. İstatistiksel ve algılama metrikleri.

	İstatistiksel Metrik		Algılama Metrikleri
	CSTest	KSTest	Lojistik regresyon
TGAN	0.994	0.769	0.404
CTGAN	0.980	0.565	0.143
CopulaGAN	0.861	0.672	0.313

Çizelge 4.8’de, istatistiksel metrikleri ve algılama metrikleri gösterilmektedir. CSTest ve KSTest değerleri dikkate alındığında, TGAN, CTGAN ve CopulaGAN'ın gerçek verilerin sürekli değerlerinden ziyade kesikli öznitelikleri daha iyi modelleyebileceği sonucuna varılmıştır. TGAN için algılama metriği, lojistik regresyon sınıflandırıcısının gerçek veriyi sentetik veriden ayırt etmesinin orta derecede olduğunu göstermektedir. TGAN ve CopulaGAN ise gerçek veriyi sentetik veriden ayırt etmesinin daha zor olduğu gözlemlenmiştir.

Şekil 4.5’de, hem gerçek hem de sentetik veriler için bir ilişkilendirme tablosu gösterilmektedir. Hangi sütunların birbiriyle ilişkili olduğu net bir şekilde anlaşılmakta ve sentetik verilerin nerede ayrıldığını göstererek modelin bu ilişkiyi modellemede yaşadığı zorluklar gösterilmektedir.



Şekil 4.5. Gerçek ve sentetik veriler arasındaki korelasyon matrisi.

Şekil 4.5’de, gerçek verilerin yanı sıra CTGAN, CopulaGAN ve TableGAN'dan üretilenin sütun bazında korelasyon gösterilmektedir. Üretilen verilerin korelasyon matrisi, gerçek verilerin korelasyonları ile karşılaştırılmıştır. Genel olarak, tüm modeller, özellikler arasındaki korelasyonları yeterince yakalayabilmektedir. Korelasyon değerleri grafiğindeki fark 0,3'te sınırlandırılmıştır, çünkü bu hedeften büyük bir sapma olarak

kabul edilmektedir. Üretilen verilerin korelasyon matrisi, gerçek verilerin korelasyonları ile karşılaştırılmaktadır. Genel olarak, TGAN modeli, sütünlar arasındaki korelasyonları yeterince yakalayabilmektedir. Görüldüğü gibi, CTGAN ve CopulaGAN ise bazı korelasyonları yakalamada zorluklarla karşılaştığı gözlemlenmiştir.

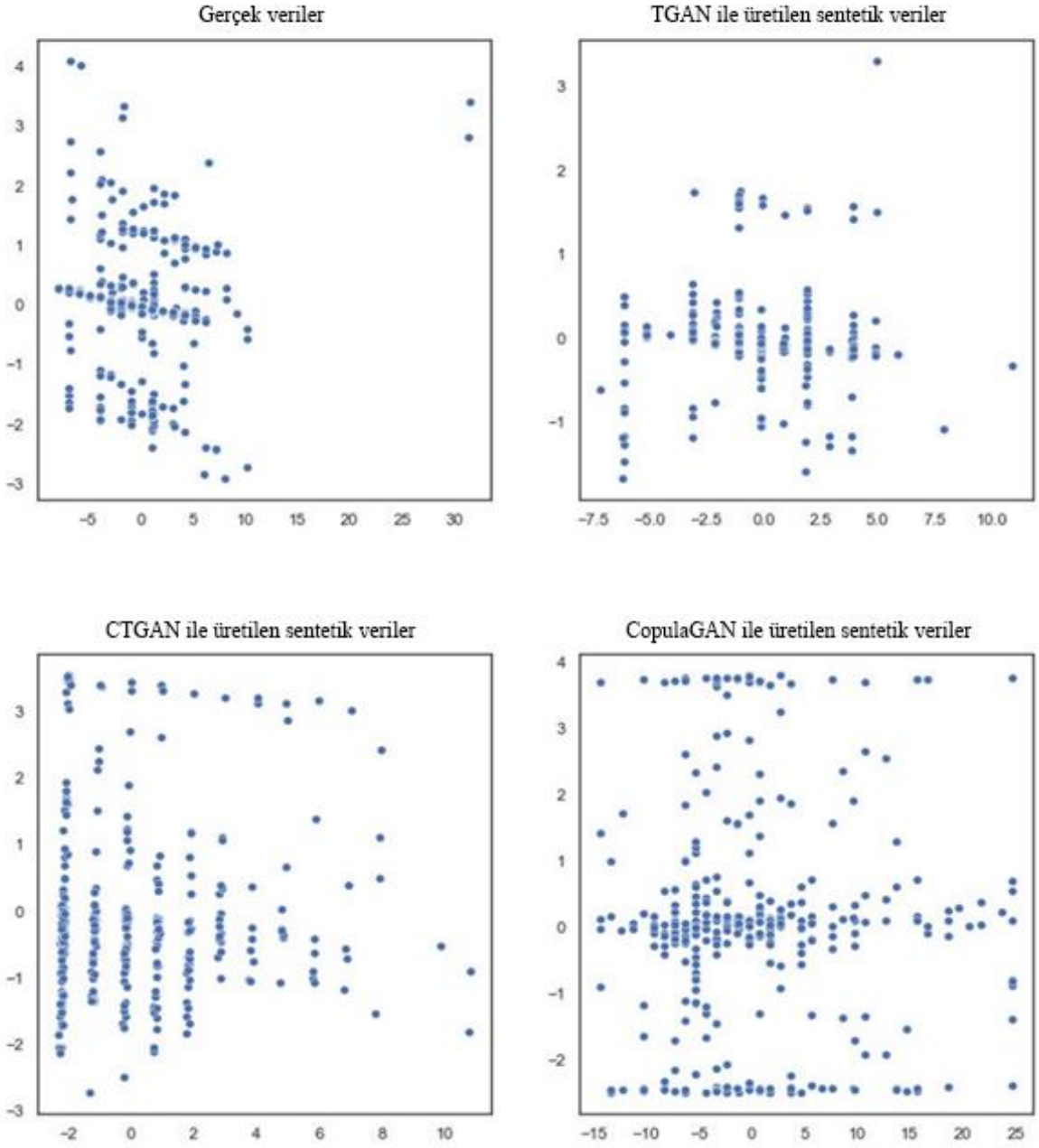
Pearson korelasyon katsayısı (Karl Pearson olarak adlandırılır), gerçek ve sentetik veri seti arasındaki doğrusal ilişkinin gücünü göstermek için kullanılmaktadır. Çizelge 4.9'da gerçek ve üretilen sentetik verilerin sütün korelasyonunu ve benzerlik puanları gösterilmektedir

Çizelge 4.9. Sütünlar Arasındaki korelasyon ve benzerlik puanı (-1< r <+1).

	TGAN	CTGAN	CopulaGAN
Sütun Korelasyonları	0.9465	0.9172	0.9000
Benzerlik Puanı	0.9022	0.8974	0.8859

Çizelge 4.9'da Sütünlar arasındaki korelasyon ve benzerlik puanlarına bakıldığında, üç modelimiz iyi sonuçlar vermektedir. Sütün korelasyonları 1'e çok yakın olduğundan iyi sonuçlar vermektedir. Bütün modellerimiz sütün korelasyonları 1'e yakın olduğundan başarılı sonuçlar göstermiştir. Benzerlik korelasyonlarına bakıldığında TGAN (0.9022) en başarılı model olarak olarak görülmektedir. CTGAN (0.8974) ve CopulaGAN (0.8859) ise benzer korelasyonları göstermektedir. Bu nedenle, gerçek ve sentetik veriler arasında iyi bir ilişkinin olduğunu göstermektedir.

Temel bileşen analizi (PCA), çok boyutlu uzaydaki bir verinin, varyansı en üst düzeye çıkaracak şekilde daha küçük bir alana izdüşümünü bulma yöntemidir. Amaç, bir vektör bir dönüşüme uğramasından sonra boyutunun değişmesinden bağımsız olarak hâlâ yönü aynı kalıyorsa PCA doğru çalışmaktadır. Şekil 4.6'da, gerçek ve sentetik veriler arasındaki temel bileşenler grafiği gösterilmiştir.



Şekil 4.6. Temel bileşenler analizi grafiği (PCA).

Şekil 4.6'da birbirini takip eden tüm temel bileşenler benzer bir yayılım ve yön takip etmekte, yani önceki bileşenle ilişkilendirilmeden kalan varyasyonu yakalamaktadır. Bu bileşenlerin yönleri denetimsiz bir şekilde tanımlanmakta, yani yanıt değişkeni(Y) bileşen yönünü belirlemek için kullanılmamaktadır. Bu nedenle TGAN, CTGAN ve CopulaGAN gerçek verilerle benzer ve denetimsiz bir yaklaşım sergilenmiştir.

Makine öğrenimi verimlilik metrikleri, sentetik veriler üzerinde bir makine öğrenimi modeli öğrenerek ve ardından gerçek veriler üzerinde değerlendirildiğinde elde ettiği puanı değerlendirerek bir makine öğrenimi problemini çözmek için gerçek verileri sentetik verilerle değiştirmenin mümkün olup olmadığını değerlendirmektedir. Makine öğrenimi metrikleri Çizelge 4.10'da verilmektedir.

Çizelge 4.10. Modeller için makine öğrenimi verimlilik metrikleri.

Makine Öğrenimi Verimlilik Metrikleri					
	Karar ağacı	AdaBoost	Lojistik Regresyon Sınıflandırıcısı	MLP sınıflandırıcı	Ortalama
TGAN	0.324	0.972	0.968	0.324	0.647
CTGAN	0.306	0.942	0.878	0.324	0.613
CopulaGAN	0.282	0.931	0.829	0.309	0.588

Çizelge 4.10'da tüm modeller, makine öğrenimi sorunlarını çözmek için gerçek verileri sentetik verilerle değiştirmenin mümkün olduğunu gösteren benzer performans göstermektedir.

Kök Ortalama Kare Hatası (RMSE) ve Ortalama Kare Hatası (MAE), sentetik verileri oluştururken ortalama model tahmin hatasını ifade etmektedir. Her iki ölçüm de 0 ile ∞ arasında değişebilmekte ve hataların yönüne kayıtsızdır. Yani negatif yönelimli puanlardır, bu da daha düşük değerlerin daha iyi olduğu anlamına gelmektedir. Çizelge 4.11'de RMSE ve MAE sonuçlarını göstermektedir.

Çizelge 4.11. RMSE ve MAE Sonuçları.

	TGAN		**CTGAN**		**CopulaGAN**	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
Gerçek Veriler	0.2410	0.1778	0.2585	0.1977	0.2519	0.1824

Deprem veri setinin modeller tarfından üretilen sentetik verilerin modellerin performans değerleri Çizelge 4.11‘de gösterilmiştir. TGAN, MAE değerinin 0’a yaklaştığı göz önüne alınan modeller arasında en başarılı model parametreleri gösterilmektedir. CTGAN ve CopulaGAN modellerin parametreleri ileştirmek için modeleri düzenlemek gerekmektedir.

5. TARTIŞMA VE SONUÇ

5.1. Sonuç

Bu çalışma, İstanbul Büyükşehir Belediyesi açık veri portalı tabanından alınan deprem veri setini, ÇÜA mimarisinin üç modeli olan TGAN, CTGAN ve CopulaGAN modelleri kullanarak sentetik veri üretimi yapılmıştır. Üretilen sentetik verilerin görsel, istatistiksel ve makine öğrenimi tabanlı metrikleriye analiz edilmiştir.

Analiz sonuçlarına göre ortalama hesaplanmasında üç modelimizde iyi sonuçlar elde etmiştir. En iyi TGAN modeli (Ortalama = 19.53) olduğunu ve deprem veri (Ortalama = 19.84) setine yakın ortalama sahip veriler üretmiştir. Kümülatif toplamları hesaplamasında en iyi TGAN modeli (toplam = 27269.58) olduğu ve deprem veri(toplam = 27269.58) setine yakın hesaplamalar oluşturmakdadır. CSTest ve KSTest değerleri dikkate alındığında, TGAN, CTGAN ve CopulaGAN'ın gerçek verilerin sürekli değerlerinden ziyade kesikli öznitelikleri daha iyi modelleyebileceği sonucuna varılmıştır. TGAN için algılma metriği, lojistik regresyon sınıflandırıcısının gerçek veriyi sentetik veriden ayırt etmesinin orta derecede olduğunu göstermektedir. CTGAN ve CopulaGAN‘da modellerinin ise gerçek veriyi sentetik veriden ayırt etmesinin zor olduğu gözlemlenmiştir. Makine öğrenimi, gerçek verileri sentetik verilerle değiştirmenin sorunları çözmede mümkün olduğunu gösteren benzer performanslar göstermektedir. Sütünlar arasındaki korelasyon ve benzerlik puanlarına bakıldığında, üç modelimiz iyi sonuçlar vermektedir. Benzerlik korelasyonlarına göre TGAN (0.9022) en başarılı model olarak olarak görülmektedir. CTGAN (0.8974) ve CopulaGAN (0.8859) ise benzer korelasyonları göstermektedir. Bu nedenle, gerçek ve sentetik veriler arasında iyi bir ilişkinin olduğunu göstermektedir.

Sonuç olarak, ÇÜA mimarisinin üç modeli olan TGAN, CTGAN ve CopulaGAN modelleri kullanarak sentetik veri üretiminde başarılı sonuçların elde edilebileceği gösterilmiştir. Bunlara ek olarak, ÇÜA'lar tarafından oluşturulan sentetik tablo verilerinin kullanımının endüstüride, görüntü işleme, finans, sağlık vb. alanlarda kullanım için ciddi bir rakip olabileceği görülmüştür. Bu sentetik tablo verilerinin kullanım kolaylığı sağlaması nedeniyle iyi sonuçlar alınabildiği anlaşılmıştır.

5.2. Tartışma

ÇÜA'da birçok yeni model gibi avantajlara ve dezavantajlara sahiptir. Dezavantajları; eğitim sırasında üretici ağ Ü ve ayırıcı ağ A'nin iyi şekilde senkronize edilmesi gerekmektedir. Ü, A'yi güncellemeden önce fazla eğitime tabii tutulmalıdır. Avantajları; Markov zincirlerine asla ihtiyaç duymamaktadır. Modelin esnek olması sayesinde birçok fonksiyon modele dahil edilebilmektedir. Ayrıca ÇÜA modeli, üst düzey öznitelikler elde ederek veri yapısını ve dağılımını modelleyebildiği ve çekişmeli eğitim yoluyla öğrenilen veri dağılımından örneklemeye olanak sağladığı için araştırmacıların ilgisini çekmekte ve ÇÜA tabanlı birçok modelin ortaya çıkmasına neden olmaktadır.

Daha iyi sonuçlar elde etmek için model eğitimi ve veri normalleştirme üzerinde daha fazla çalışabilmektedir.

Veri türlerinin sayısını artıran yöntemler araştırılabilmekte, tarihler ve sıralı veriler, araştırma için uygun yönlerdir.

ÇÜA'ların zaman serisi verilerini, çarpık ağır kuyruklu dağılımlara sahip verileri ve sınırlı sayısal verileri işlemek için nasıl tasarlanabileceği araştırılmalıdır. Ayrıca ayrık sayısal verilerin ve sıralı verilerin nasıl sentezleneceği ileriki çalışmalarda araştırılması gereken konular arasındadır.

KAYNAKLAR

Aas, K., Czado, C., Frigessi, A., Bakken, H., 2009. Pair-copula constructions of multiple dependence. ***Insurance: Mathematics and Economics*, 44** (2): 182-198.

Bourou, S., El Saer, A., Velivassaki, T. H., Voulkidis, A., Zahariadis, T., 2021. A Review of Tabular Data Synthesis Using GANs on an IDS Dataset. ***Information*, 12** (9): 375.

Brenninkmeijer, B., De Vries, A., Marchiori, E., Hille, Y., 2019**.** ***On the Generation and Evaluation of Tabular Data Using GANs*** (doctoral thesis, unpublished). Radboud University, Institute for Science in Society, Nijmegen, The Netherlands.

Brenninkmeijer, B., 2021. Welcome to table evaluator's documentation. ***https://baukebrenninkmeijer.github.io/table-evaluator/*.** Table Evaluator, Holland. Erişim tarihi: 21.12.2021.

Bringas, P. G., Santos, I., 2010. Bayesian Networks for Network Intrusion Detection. 14. ***Bayessian Network*** (Editor: A. Rebai). InTech, London. 229.

Chang, B., Pan, S., Joe, H., 2019. Vine copula structure learning via Monte Carlo tree search. ***22ND International Conference on Artificial Intelligence and Statistics***, **89**: 353-361.

Charlier, J., Singh, A., Ormazabal, G., State, R., Schulzrinne, H., 2019. SynGAN: Towards generating synthetic network attacks using GANs. ***https://www.arxiv-vanity.com/papers/1908.09899*.** arXiv Vanity, New York. Erişim tarihi: 12.02.2008.

Che, Z., Cheng, Y., Zhai, S., Sun, Z., Liu, Y., 2017. Boosting deep learning risk prediction with generative adversarial networks for electronic health records. ***In 2017 IEEE International Conference on Data Mining (ICDM).*** 18-21 November 2017, New Orleans. 787-792.

Choi, E., Biswal, S., Malin, B., Duke, J., Stewart, W. F., Sun, J., 2017. Generating multi-label discrete patient records using generative adversarial networks. ***in Machine Learning for Healthcare Conference***. 18-19 August 2017, Massachusetts. 286-305.

Elidan, G., 2010. Copula Bayesian Networks. ***Proceedings of the 23rd International Conference on Neural Information Processing Systems.*** 6 December 2010, United States. 559-567.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Machine learning basics, Chap. 5. ***Deep Learning*** (Editor: D. Jackson). MIT Press, London. 95.

Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative Adversarial Nets. ***Proceedings of the 27th International Conference on Neural Information Processing Systems***, **2**: 2672-2680.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A., 2017. Improved training of wasserstein gans. ***Advances in Neural Information Processing Systems***, **30**: 5767-5777.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. ***MIT Press, 9*** (8): 1735-1780.

Hu, W., Tan, Y., 2017. Generating adversarial malware examples for black-box attacks based on GAN. ***https://www.arxiv-vanity.com/papers/1702.05983/*.** arXiv Vanity, New York. Erişim tarihi: 12.02.2008.

Latif, S., Usman, M., Manzoor, S., Iqbal, W., Qadir, J., Tyson, G., Crowcroft, J., 2020. Leveraging data science to combat covid-19: A comprehensive review. ***IEEE Transactions on Artificial Intelligence,*** **1** (1): 85-103.

Kingma, D. P., Welling, M., 2019. Growth and yield responses of Italian ryegrass (*Lolium multiflorum*) to diclofop-methyl and ozone. ***DBLP***, **12** (4): 307-392.

Kinney, S. K., Reiter, J. P., Reznek, A. P., Miranda, J., Jarmin, R. S., Abowd, J. M., 2011. Towards unrestricted public use business microdata: The synthetic longitudinal business database. ***International Statistical Review,*** **79** (3): 362-384.

Kohli, M. D., Summers, R. M., Geis, J. R., 2017. Medical image data and datasets in the era of machine learning—whitepaper from the 2016 C-MIMI meeting dataset session. ***Journal of Digital Imaging,*** **30** (4): 392-399.

Krizhevsky, A., Sutskever, I., Hinton, G. E., 2012. Imagenet classification with deep convolutional neural networks. ***Advances in Neural Information Processing Systems.*** 3 – 6 December 2012, United States. 1097-1105.

Li, Z., Qin, Z., Huang, K., Yang, X., Ye, S., 2017. Activity and Selectivity in the Field, Chap. 5. ***Neural Information Processing*** (Editor: K. Kim, M. E. Aminanto, H. C. Tanuwidjaja). Springer, Singapur. 858-866.

Lin, Z., Khetan, A., Fanti, G., Oh, S., 2020. Pacgan: The power of two samples in generative adversarial networks. ***IEEE Journal on Selected Areas in Information Theory,*** **1** (1): 324-335.

Lin, Z., Shi, Y., Xue, Z., 2018. IDSGAN: Generative Adversarial Networks for Attack Generation against Intrusion Detection. ***https://www.arxiv-vanity.com/papers/1809.02077/.*** arXiv Vanity, New York. Erişim tarihi: 12.02.2008.

Liu, S., Wang, T., Bau, D., Zhu, J. Y., Torbala, A., 2020. Generative Deep Learning for Internet of Things Network Traffic Generation. ***In 2017 IEEE International Conference on Data Mining (ICDM).*** 14-19 June 2020, New Orleans. 14286-14295.

Mottini, A., Lheritier, A., Acuna-Agost, R., 2018. Airline passenger name record generation using generative adversarial networks. ***https://deepai.org/publication/airline-passenger-name-record-generation-using-generative-adversarial-networks.*** DeepAI, California. Erişim tarihi: 05.06.2021.

Murphy, K., 2012. The social pillar of sustainable development: a literature review and framework for policy analysis. ***Sustainability: Science, Practice and Policy,*** **8** (1): 15-29.

Park, N., Mohammadi, M., Gorde, K., Jajodia, S., Park, H., Kim, Y., 2018. Data synthesis based on generative adversarial networks. ***Proceedings of the VLDB Endowment***, **11** (10): 1071-1083.

Chandar, S., Khapra, M. M., Larochelle, H., Ravindran, B., 2016. Correlational neural networks. ***Neural Computation***, **28** (2): 257-285.

Patki, N., Wedge, R., Veeramachaneni, K., 2016. The synthetic data vault. ***2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA).*** 17-19 October 2016, Canada. 399-410.

Patki, N., Wedge, R., Veeramachaneni, K., 2018. CopulaGAN Model. ***https://sdv.dev/SDV/user_guides/single_table/copulagan.***. The Synthetic Data Vault, Massachusetts. Erişim tarihi: 21.12.2021.

Patki, N., Wedge, R., Veeramachaneni, K., 2018. Synthetic Data Evaluation-Single Table Metrics. ***https://sdv.dev/SDV/user_guides/evaluation/single_table_metrics.*** The Synthetic Data Vault, Massachusetts. Erişim tarihi: 21.12.2021.

Radford, A., Metz, L., Chintala, S., 2016. Unsupervised representation learning with deep convolutional generative adversarial networks. ***Under Review as a Conference Paper at ICLR 2016.*** 2-4 May 2016, Puerto Rico. 1-16.

Rai, K., Devi, M. S., Guleria, A., 2016. Decision tree based algorithm for intrusion detection. ***International Journal of Advanced Networking and Applications***, **7** (4): 2828.

Reiter, J. P., 2005 . Releasing multiply imputed, synthetic public use microdata: an illustration and empirical study. ***Journal of the Royal Statistical Society: Series A (Statistics in Society),*** **168** (1): 185-205.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. ***The journal of Machine Learning Research,*** **15** (1): 1929-1958.

Teng, S., Wu, N., Zhu, H., Teng, L., Zhang, W.,2017. SVM-DT-based adaptive and collaborative intrusion detection. ***IEEE/CAA Journal of Automatica Sinica,*** **5** (1): 108-118.

Wason, R., 2018. Deep learning: Evolution and expansion. ***Cognitive Systems Research***, **52**: 701-708.

Xu, L., Veeramachaneni, K., 2018. Synthesizing Tabular Data using Generative Adversarial Networks. ***https://www.arxiv-vanity.com/papers/1811.11264/.*** arXiv Vanity, New York. Erişim tarihi: 12.11.2021.

Xu, L., Skoularidou, M., Cuesta-Infante, A., Veeramachaneni, K., 2019. Modeling Tabular Data using Conditional GAN. ***NIPS'19: Proceedings of the 33rd International Conference on Neural Information Processing Systems.*** 8-14 December 2019, Kanada. 7335–7345.

Yi, X., Walia, E., Babyn, P., 2019. Generative adversarial network in medical imaging: A review. ***Medical Image Analysis,*** **58**: 101552.

ÖZ GEÇMİŞ

İlk, orta ve lise eğitimini Diyarbakır'da tamamladı. 2008 yılında Atatürk Üniversitesi Bilgisayar Teknolojileri ve Programlama bölümünden, 2013 yılında Van Yüzüncü Yıl Üniversitesi Bilgisayar Öğretmenliği ve Teknolojileri Eğitimi bölümünden mezun oldu. 2017 yılında Hacettepe Üniversitesi Bilgisayar Mühendisliği bölümünden terk. 2012- 2022 yılları arasında Van Yüzüncü Üniversitesi Bilgisayar Bilimleri Araştırma ve Uygulama Merkez müdürlüğünde, 2019 – 2022 yılları arasında Van Yüzüncü Üniversitesi Kurumsal iletişim koordinatörlüğünde grafik-tasarım ve içerik yöneticisi olarak görev yapmaktadır. Evli ve Ahmet Ömür'ün babasıdır.

VAN YÜZÜNCÜ YIL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
LİSANSÜSTÜ TEZ ORİJİNALLİK RAPORU

Tarih: 24/01/2022

Tez Başlığı / Konusu: ÇEKİŞMELİ ÜRETİCİ AĞLAR (ÇÜA) İLE SENTETİK VERİ ÜRETME

Yukarıda başlığı/konusu belirlenen tez çalışmamın Kapak sayfası, Giriş, Ana bölümler ve Sonuç bölümlerinden oluşan toplam 43 sayfalık kısmına ilişkin, 24/01/2022 tarihinde şahsım/tez danışmanım tarafından Turnitin intihal tespit programından aşağıda belirtilen filtreleme uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı %5 (Yüzde Beş) dir.

Uygulanan filtreler aşağıda verilmiştir:
- Materyal ve yöntem hariç,
- Kaynaklar hariç,
- Tezden çıkan yayınlar hariç,
- 7 kelimeden daha az örtüşme içeren metin kısımları hariç (Limit inatch size to 7 words)

Van Yüzüncü Yıl Üniversitesi Lisansüstü Tez Orijinallik Raporu Alınması ve Kullanılmasına İlişkin Yönergeyi inceledim ve bu yönergede belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini bilgilerinize arz ederim.

24/01/2022

Adı Soyadı: Hayrullah URCAN

Öğrenci No: 17910002116

Anabilim Dalı: İstatistik

Programı: İstatistik

Statüsü: ■ Yüksek Lisans □ Doktora

DANIŞMAN ONAYI
UYGUNDUR

Dr. Öğr. Üyesi Murat CANAYAZ

ENSTİTÜ ONAY
UYGUNDUR