Çevrimiçi Ödemelerde Sahtekarlığın Anlaşılması

Makine Öğrenmesi
BM5702

Hakan SÖNMEZ
502231006

ÖZET

Çevrimiçi ödemeler, teknolojinin hızla ilerlemesiyle birlikte hayatımızın önemli bir parçası haline gelmiştir. Artık birçok insan, çevrimiçi alışveriş yaparken veya faturalarını öderken geleneksel yöntemler yerine dijital ödeme yöntemlerini tercih etmektedir. Çevrimiçi ödemelerin bu kadar yaygınlaşmasının sebebi kolaylık ve hız yatmaktadır.
Yaygınlaşan çevrimiçi ödemelerle birlikte bu işlemlerden haksız kazanç sağlamak isteyenleri doğurmuştur. Bu çalışmada çevrimiçi ödemelerdeki sahtekarlıkların anlaşılması için makine öğrenmesi algoritmaları kullanılmıştır. Çalışma sonunda başarı oranları

1.GİRİŞ

Çevrimiçi ödemelerde sahtekarlık, internet üzerinde yapılan ödemeler sırasında dolandırıcıların, kullanıcıları aldatmak, kişisel ve finansal bilgileri ele geçirmek veya maddi kazanç elde etmek amacıyla kullandığı hileli ve yanıltıcı yöntemlerin bir bütünüdür. Bu tür sahtekarlık vakaları, çevrimiçi alışveriş siteleri, bankalar, ödeme platformları ve diğer dijital ödeme hizmetlerinin hedef alınmasıyla gerçekleşebilir.

Çevrimiçi ödemelerde sahtekarlık vakaları şu şekillerde ortaya çıkabilir:

Phishing (Kimlik Avı): Dolandırıcılar, sahte e-posta, mesaj veya telefon aramaları yoluyla kişisel ve finansal bilgileri elde etmeye çalışır. Örneğin, sahte banka e-postaları veya popüler e-ticaret sitelerinin taklitleri kullanılarak kullanıcılar aldatılır ve kişisel bilgileri (kullanıcı adı, şifre, kredi kartı bilgileri vb.) vermesi sağlanır.

Sahte Web Siteleri: Dolandırıcılar, meşru bir kuruluşun web sitesinin kopyasını oluşturarak kullanıcıları yanıltmaya çalışır. Kullanıcılar, sahte web sitesine girdiklerinde kredi kartı bilgilerini veya diğer ödeme ayrıntılarını girerler ve bu bilgiler dolandırıcılara ulaşır.

Hesap Hırsızlığı: Dolandırıcılar, kullanıcıların çevrimiçi hesaplarını ele geçirerek kişisel bilgileri ve finansal verileri kullanır. Zayıf şifreler, kırılabilecek güvenlik önlemleri veya phishing saldırıları aracılığıyla hesap bilgilerine erişebilirler.

Sahte Satışlar ve İade İstekleri: Dolandırıcılar, sahte ürün satışı veya iade talepleri gibi taktikler kullanarak insanları aldatmaya çalışır. Örneğin, ödeme yapıldıktan sonra ürün göndermeyebilirler veya sahte bir ürün gönderebilirler.

1.1 Problem Tanımı

Günümüzde son kullanıcılar için çevrimiçi alışveriş git gide artan öneme sahip olmakla beraber bu satışı yapan firmalar için ödemelerde sahtekarlığın anlaşılması ve ödemenin o anda reddedilmesi firma için büyük önem arz etmektedir. Çünkü anlaşılmayan sahtekarlık firmanın iade ve iptal süreçleriyle uğraşması ve postalanan ürünün geri alınamaması ya da son kullanıcının zarar etmesi sonuçlarını doğurmaktadır. Bunların engellenmesi çevrimiçi ödeme platformları için büyük bir sorundur.

2020 yılında e-ticaret siteleri çevrimiçi ödemelerdeki sahtekarlıklar yüzünden 20 milyar dolardan fazla zarar etti. Bu sayı 2022 yılında 41 milyar dolar oldu ve 2023 yılında ise 48 milyar doları aşması beklenmektedir.

E-ticaret sitelerin bu sahtekarlıklarla klasik yordamlarla ya da insan gücüyle başa çıkması imkansız bir görevdir.

1.2 Probleme Çözüm Önerisi

Günümüz dünyasında milyonlarca insanın artık çevrimiçi alışveriş yaptığı düşünülürse ve bu sayının hiç bir zaman azalmayıp her zaman artacağı da eklenirse bu problemin çözümü ancak yapay zeka tarafından yapılabileceği analşılır. Problemin çözümü için kaggle üzerinden bulunan veri setinde KNN, Naive Bayes, Lojistik Regresyon, Karar Ağaçları, Rastgele Orman, Gradient Arttırma ve Karar Destek Sistemleri algoritmalarıyla modeller denenmiş ve sonuçları karşılaştırmalı olarak gösterilmiştir.

2.KURAMSAL TEMELLER VE KAYNAK ÖZETLERİ

Bu çalışmada çevrimiçi ödemelerde sahtekarlığın anlaşılması adına gerçekleştirilen literatür taramasında ulusal tez arşivinde yüzden fazla tezin on kadarı incelenip aşağıda en yakın bulunanlardan bahsedilmiştir.

[1] çalışmasında veri kümesi olarak, Eylül 2013'te Avrupalı kart sahiplerinin kredi kartı işlemlerinin bulunduğu veriler kullanılarak yapılan bu çalışmada veri kümesi üzerinde derin öğrenme, Rastgele Orman ve sınıflandırıcı yığını yöntemleri kullanılmış ve bu yöntemler karşılaştırılmıştır. Bu veri kümesinin dengesiz olması sebebiyle örnekleme yöntemlerinden de faydalanılmıştır. Çalışma sonucunda derin öğrenme ile 0.963, rastgele orman yöntemi ile 0.956, sınıflandırıcı yığını ile 0.979 AUC değeri elde edilmiştir.

[2] çalışması da aynı veri setiyle yapılmış olup bir dizi ön işlemden sonra geliştirilen modeller ve F1 skorları sırasıyla şu şekildedir. Decision Tree ile 0.99 K nearest neighbours ile 0.102, Logistic Regression ile 0.66, Support Vector 0.00002, Random Forest ile 0.75 ve XGBoost Classifier ile 0.85 başarı sağlanmıştır.

[3] Aynı veri setiyle ilgilenen bu çalışmada da cross validation, yeniden örnekleme ve öznitelik seçme yöntemleri ile makine öğrenmesi algoritmaları kullanılmıştır. Duyarlılık metriğinde LR, XGB, LGBM, RF ve SVM metotlarının sonuçları sırasıyla şöyledir: %91, %90, %90, %88 ve %88. Kesinlik metriğinde ise RF, XGB, LGBM, SVM ve LR metotlarının sonuçları şöyledir: %99, %98, %98, %98, %97. Doğruluk metriğinde XGB, LGBM, LR, RF ve SVM metotlarının sonuçları şöyledir: %94, %94, %94, %93, %93.

3. Veri Seti

Modelleri oluşturmak için kullanılan veri seti kaggle üzerinden açık kaynaklı olarak bulunmuştur.
https://www.kaggle.com/datasets/jainilcoder/online-payment-fraud-detection

Veri seti 6362620 işlemden oluşmaktadır ve bunların sadece 8213 adeti fraud olarak işaretlenmiştir. Bu haliyle bu veri seti bir dengesiz veri setidir.
Her bir satır olan işlemin eski ve yeni bakiye miktarı hangi hesaplar arası olduğu ve kaç adımda olduğu bilgilerini tutmaktadır. Veri setinin özniteliklerinin açıklamaları aşağıdaki gibidir.

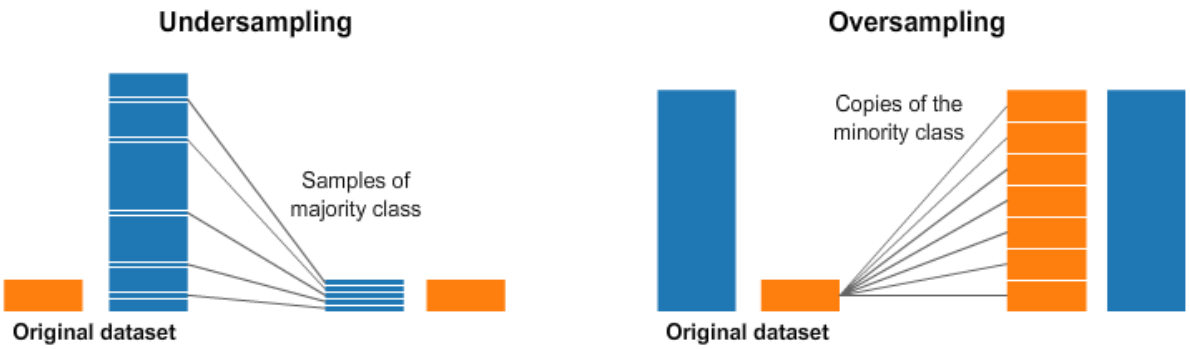
Sütun Adı	Açıklaması
step	1 adımın 1 saate eşit olduğu bir zaman birimini temsil eder.
type	İşlemin tipi
amount	İşlemin miktarı
nameOrig	Müşternin işleme başlaması
oldbalanceOrg	İşlemden önce bakiye
newbalanceOrig	İşlemden sonra bakiye
nameDest	İşlemin alıcısı
oldbalanceDest:	İşlemden önce alıcının ilk bakiyesi
newbalanceDest	İşlemden sonra alıcının ilk bakiyesi
isFraud	Sahtecilik mi değil mi
isFlaggedFraud	Sahtecilik olarak işaretlenebilmiş

3.1 Dengesiz Veri Seti

Dengesiz veri seti sınıflandırma problemlerinde görülür ve sınıf dağılımlarının birbirine yakın olmadığı durumlarda ortaya çıkar. Problem çoğunluğa sahip sınıfın azınlık sınıfını domine etmesinden kaynaklanır. Oluşturulan model çoğunluğa sahip sınıfa yakınlık gösterir bu da azınlık sınıfının kötü sınıflandırılmasına sebep olur.

3.2 Dengesiz Veri Setine Uygulanan Çözümler

Dengesiz veri setiyle başa çıkmak uygulanan metodların başında yeniden örnekleme (resampling) gelir. Resampling, zınlık sınıfına yeni örnekler ekleyerek veya çoğunluk sınıfından örnekler çıkarılarak veri setinin daha dengeli hale getirilmesidir. Resampling undersampling ve oversampling olarak ikiye olarak ayrılır.



Oversampling ve Undersampling görselleştirilmiştir.

3.2.1 Undersampling

Çoğunluk sınıfına ait örneklerin çıkarılmasıyla veri setini dengeleme tekniğidir. 4 alt teknikte incelenebilir.

1. **Random Undersampling:** Çıkarılan örnekler rastgele seçilir. Büyük veri setine sahipseniz bu tekniği kullanabilirsiniz. Rastgele seçimden dolayı bilgi kaybı yaşanabilir.
2. **NearMiss Undersampling**: Bilgi kaybını önler. KNN algoritmasına dayanır. Çoğunluk sınıfına ait örneklerin azınlık sınıfına ait örneklerle olan uzaklığı hesaplanır. Belirtilen k değerine göre uzaklığı kısa olan örnekler korunur.
3. **Undersampling (Tomek links)** :Farklı sınıflara ait en yakın iki örneğin arasındaki çoğunluk sınıfının örnekleri kaldırılarak, iki sınıf arasındaki boşluk arttırılır.
4. **Undersampling (Cluster Centroids)** : Önemsiz örneklerin veri setinden çıkarılmasıdır.Örneğin önemli veya önemsiz olduğu kümelemeyle belirlenir.

3.2.2 Oversampling

Azınlık sınıfına ait örneklerin kopyalanmasıyla veri setini dengeler. 2 alt teknikte incelenebilir.

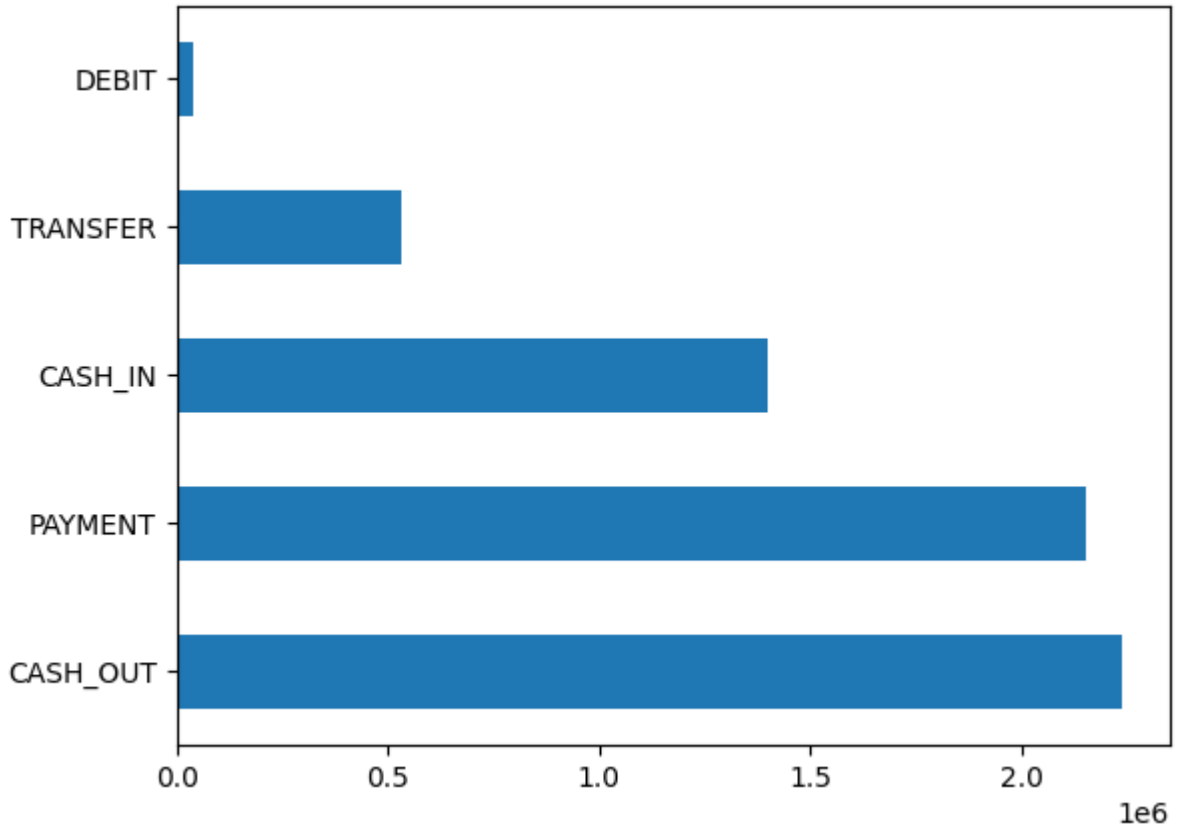
1. **Random Oversampling**: Azınlık sınıfından rastgele seçilen örneklerin eklenmesiyle veri setinin dengelenmesidir. Veri setiniz küçükse bu teknik kullanılabilinir. Overfitting'e neden olabilir.
2. **SMOTE Oversampling:** Ezberlemeyi önlemek için azınlık sınıfından sentetik örnekler oluşturulması.

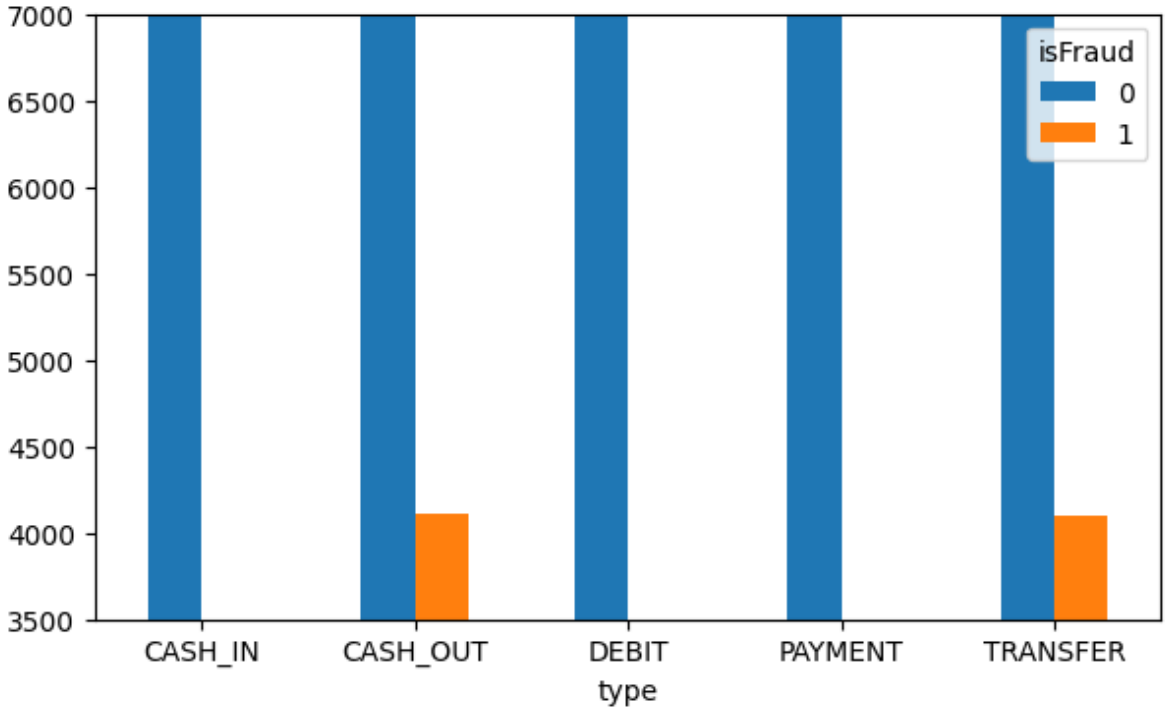
3.3 Veri Setine Uygulanan Ön İşlemler

Veri setini daha iyi anlamak için öncelikle describe ile sayısal özniteliklerin incelenmesi yapılmıştır.

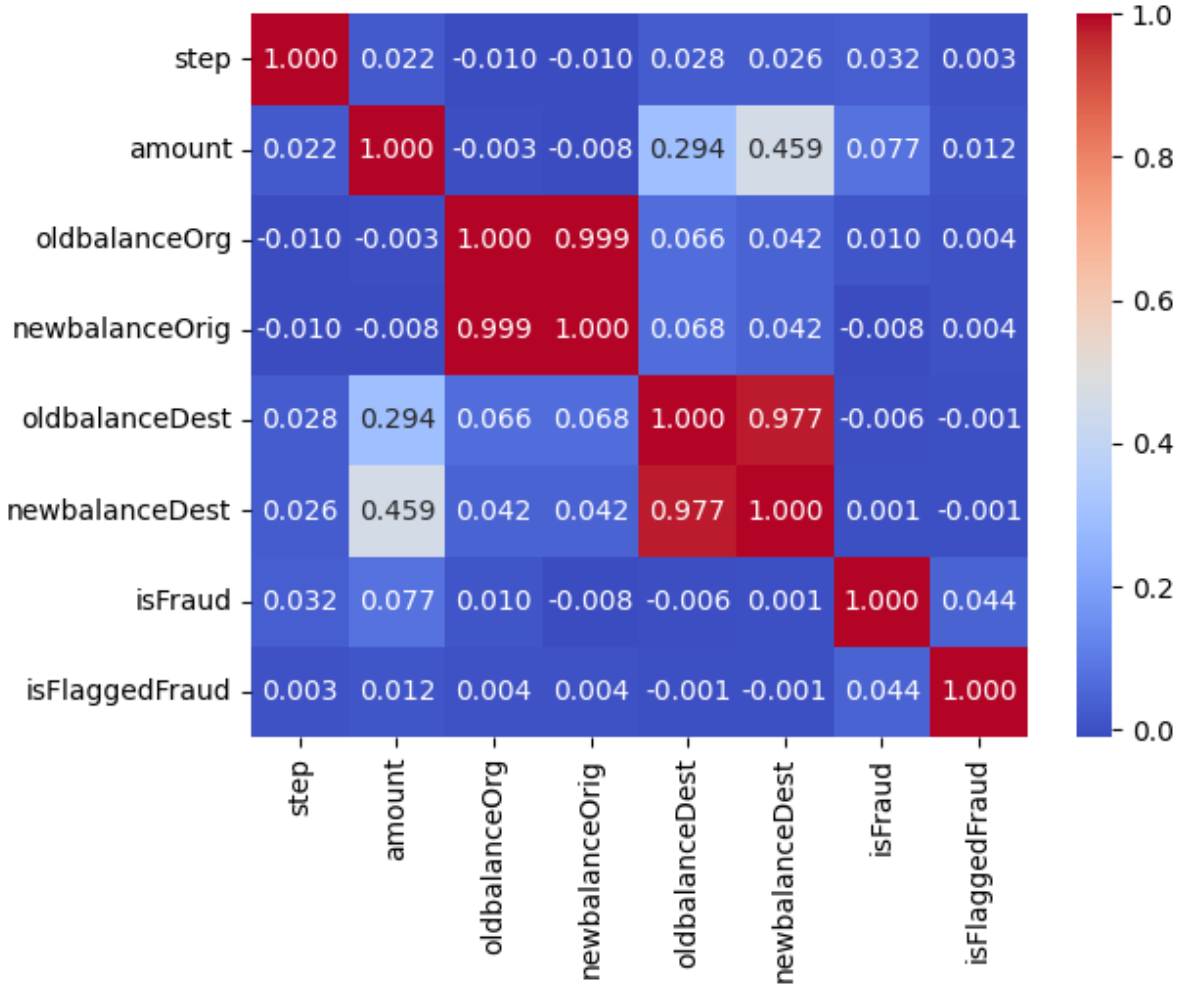
	count	mean	std	min	25%	50%	75%	max
step	6.36e+06	2.43e+02	1.42e+02	1.0	156.00	239.00	3.35e+02	7.43e+02
amount	6.36e+06	1.80e+05	6.04e+05	0.0	13389.57	74871.94	2.09e+05	9.24e+07
oldbalanceOrg	6.36e+06	8.34e+05	2.89e+06	0.0	0.00	14208.00	1.07e+05	5.96e+07
newbalanceOrig	6.36e+06	8.55e+05	2.92e+06	0.0	0.00	0.00	1.44e+05	4.96e+07
oldbalanceDest	6.36e+06	1.10e+06	3.40e+06	0.0	0.00	132705.66	9.43e+05	3.56e+08
newbalanceDest	6.36e+06	1.22e+06	3.67e+06	0.0	0.00	214661.44	1.11e+06	3.56e+08
isFraud	6.36e+06	1.29e-03	3.59e-02	0.0	0.00	0.00	0.00e+00	1.00e+00
isFlaggedFraud	6.36e+06	2.51e-06	1.59e-03	0.0	0.00	0.00	0.00e+00	1.00e+00

Veri setinde type niteliğine göre işlem sayıları gösterilmiştir.





Tiplere göre Fraud sayıları maksimum 7000 işlem özelinde gösterilmiştir.



Öznitelikler arasındaki korelasyon incelenmiştir ve buna göre oldbalanceOrg ve newbalanceOrig arasında kuvvetli pozitif korelasyon aynı zamanda oldbalanceDest ve newbalanceDest arasında güçlü pozitif korelasyon bulunmuştur. Bu sütunlardan veri setimizden düşürülerek gereksiz işlem maliyetinden kaçınılmıştır.

Ayrıca veri setimizi modellemeden önce balance_diff adında yeni bir sütun oldbalanceOrg - newbalanceOrig eşitliği olacak şekilde eklenmiştir

Veri setimizdeki type sütununa göre isFraud değerlerinin sayısal karşılıklarına bakıldığında sadece CASH_OUT ve TRANSFER tipinde fraud işlem gözükmektedir. Bu bilgiye göre ikinci tip modellerimizde CASH_IN, DEBIT ve PAYMENT tiplerindeki satırlar düşürülerek sadece CASH_OUT ve TRANSFER satırları kullanılmıştır.

isFraud	0	1
type		
CASH_IN	1399284	0
CASH_OUT	2233384	4116
DEBIT	41432	0
PAYMENT	2151495	0
TRANSFER	528812	4097

4. Makine Öğrenmesi Algoritmaları

Makine öğrenmesi, yapay zekanın bir alt dalı olan, geçmiş deneyimlerden insan müdahalesi olmadan öğrenme yeteneğine sahip, gelecek çıktıları tahmin edebilen metotlardır. Hangi durumda ne yapacağı geleneksel programlama yönteminde olduğu gibi açıkca bildirilmeden, kendisinin verilerdeki örüntü, karakteristik aracılığıyla öğrendiği yöntemlerdir.

Bu çalışmada 6 adet makine öğrenme algoritması kullanılarak çalışılmıştır.

4.1 K Nearest Neighbour (K-En Yakın Komşu)

KNN, içerisinde tahmin edilecek değerin bağımsız değişkenlerinin oluşturduğu vektörün en yakın komşularının hangi sınıfta yoğun olduğu bilgisi üzerinden sınıfını tahmin etmeye dayanır.

KNN algoritması iki temel değer üzerinden tahmin yapar;

1. **Distance (Uzaklık)**: Tahmin edilecek noktanın diğer noktalara uzaklığı hesaplanır. Bunun için Minkowski uzaklık hesaplama fonksiyonu kullanılır.
2. **K (komuşuluk sayısı)**: En yakın kaç komşu üzerinden hesaplama yapılacağını söyleriz. K değeri sonucu direkt etkileyecektir. K 1 olursa overfit etme olasılığı çok yüksek olacaktır. Çok büyük olursa da çok genel sonuçlar verecektir. Bu sebeple optimum K değerini tahmin etmek problemin asıl konusu olarak karşımızda durmaktadır.

4.2 Naive Bayes (Toy Bayes)

Naive Bayes sınıflandırıcısının temeli Bayes teoremine dayanır. lazy (tembel) bir öğrenme algoritmasıdır aynı zamanda dengesiz veri kümelerinde de çalışabilir. Algoritmanın çalışma şekli bir eleman için her durumun olasılığını hesaplar ve olasılık değeri en yüksek olana göre sınıflandırır.

4.3 Logistic Regression (Lojistik regresyon)

Lojistik regresyon, iki veri faktörü arasındaki ilişkileri bulmak için matematikten yararlanan bir veri analizi tekniğidir. Lojistik regresyon, daha sonra diğerine dayalı bu faktörlerden birinin değerini tahmin etmek için bu ilişkiyi kullanır. Tahminin genellikle evet ya da hayır gibi sınırlı sayıda sonucu vardır.

4.4 Decision Tree (Karar ağaçları)

Karar ağaçları, sınıflama, özellik ve hedefe göre karar düğümleri (decision nodes) ve yaprak düğümlerinden (leaf nodes) oluşan ağaç yapısı formunda bir model oluşturan bir sınıflandırma yöntemidir. Karar ağacı algoritması, veri setini küçük ve hatta daha küçük parçalara bölerek geliştirilir. Bir karar düğümü bir veya birden fazla dallanma içerebilir. İlk düğüme kök düğüm (root node) denir. Bir karar ağacı hem kategorik hem de sayısal verilerden oluşabilir.

4.5 Random Forest (Rassal orman)

Rassal orman, hiper parametre kestirimi yapılmadan da iyi sonuçlar vermesi hem regresyon hem de sınıflandırma problemlerine uygulanabilir olmasından dolayı popüler makine öğrenmesi modellerinden biridir. Rassal orman modeli varsayılan olarak 100 adet alt karar ağacı oluşturup sonucu bunların sonucuna göre vererek çalışmaktadır.

4.6 Gradient Boosting

Boosting, zayıf öğrenicileri(weak learner) güçlü öğreniciye(strong learner) dönüştürme yöntemidir. Bunu iterasyonlar ile aşamalı olarak yapar. Boosting algoritmaları arasındaki fark genellikle zayıf öğrenicilerin eksikliğini nasıl tanımladıklarıdır.

Gradient Boosting'de öncelikli olarak ilk yaprak(initial leaf) oluşturulur. Sonrasında tahmin hataları göz önüne alınarak yeni ağaçlar oluşturulur. Bu durum karar verilen ağaç sayısına ya da modelden daha fazla gelişme kaydedilemeyinceye kadar devam eder.

5. Performans Ölçme Araçları

Sınıflandırma problemlerinde modellerin performansını ölçmek için öncelikle True Pozitif (TP), False Pozitif (FP), True Negatif (TN) ve False Negatif (FN) değerlerinin bilinmesi gerekmektedir.

- **True Positive (TP)**: Asıl sınıfı pozitif (dolandırıcılık) olan bir verinin pozitif olarak tahmin edilmesi durumudur.
- **False Positive (FP)**: Asıl sınıfı negatif (normal) olan bir verinin pozitif olarak tahmin edilmesi durumudur.
- **True Negative (TN):** Asıl sınıfı negatif (normal) olan bir verinin negatif olarak tahmin edilmesi durumudur.
- **False Negative (FN)**: Asıl sınıfı pozitif (dolandırıcılık) olan bir verinin negatif olarak tahmin edilmesi durumudur.

Sınıflandırma problemlerinde modellerin performansını ölçmek için aşağıdaki metrikler kullanılır.

- **Precision (Kesinlik):** Pozitif olarak tahmin edilenlerin ne kadarının gerçekte pozitif olduğunu gösterir. Eğer precision düşük ise çok sayıda hatalı pozitif olduğunu ifade eder.

- **Recall (Duyarlılık)**: Pozitif olarak tahmin etmemiz gereken değerlerin ne kadarını pozitif tahmin ettiğimizi gösterir.Eğer recall düşük ise çok sayıda yanlış negatif olduğunu ifade eder.

- **F1 score** : Precision ve Duyarlılık değerlerinin harmonik ortalamasını göstermektedir.

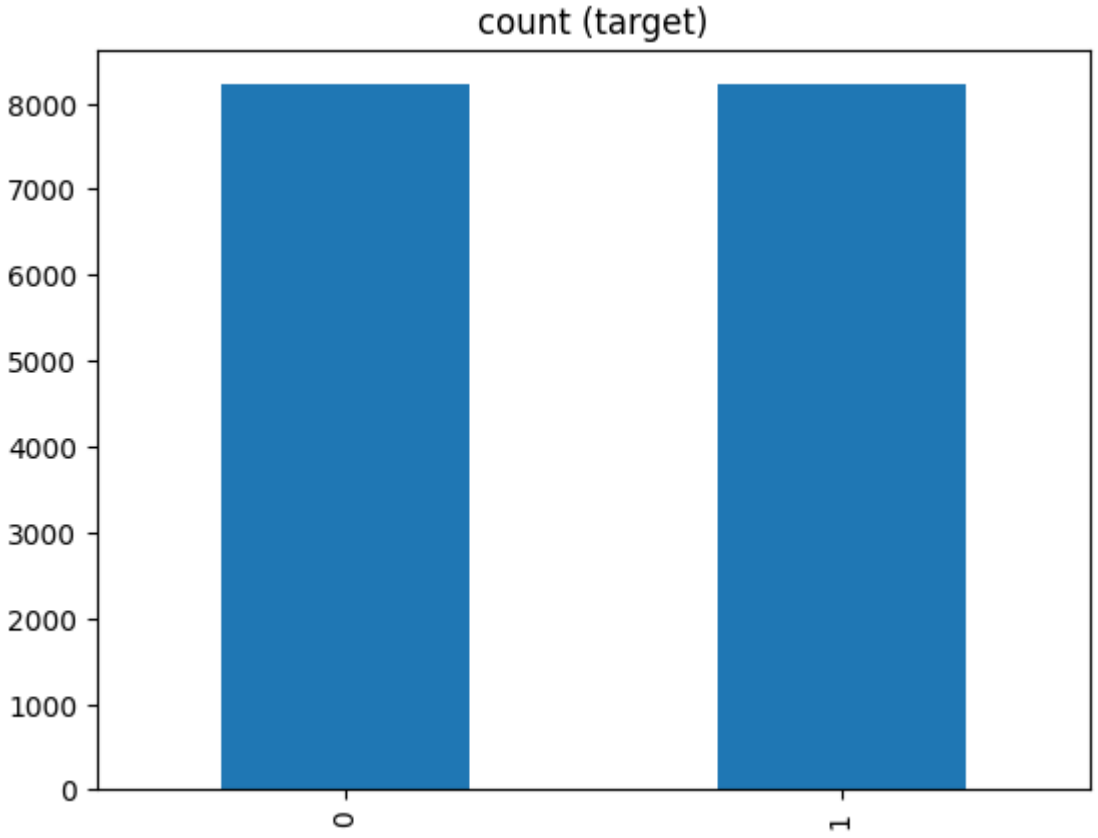
- **ROC curve**: Farklı sınıflandırma eşiklerinde gerçek pozitif oran ve yanlış pozitif oran eğrisidir. (0,0) ‘da başlar ve (1,1)’ de biter. İyi bir model, 0'dan 1'e hızla giden bir eğri üretir.

- **AUC (Area under the ROC curve)**: ROC eğrisini tek bir sayı ile özetler. En iyi değer 1.0, en kötü değeri 0.5'dir.

6. Sonuçlar

Modellerin daha iyi sonuç vermesi ve GridSearchCV sınıfından faydanılmıştır. Bu sınıf verilen modeli verilen hiper parametre değerleri ve cross validation değeriyle kartezyen çarpımı sayısında model oluşturarak en iyi parametre ve en iyi veri sınıfını göstermektedir. Bu çalışmada modellerin tamamı önce GridSearchcv ile en iyi hiper parametreleri makul süre içerisinde bulunup kullanılmıştır.

Undersampling yapılan modellerimizde toplamda 16426 satır veri kullanılmıştır. Bunlardan 8213 tanesi fraud ve 8213 tanesi fraud olmayan veridir.



Verilerin dağılımı gösterilmiştir.

Sonuçlar iki başlık altında incelenmiştir. Birinci sonuçlar undersampling yapılarak GridSearchCV'den çıkan en iyi modelde çalıştırılara elde edilmiştir. İkinci modeller ise veri setindeki CASH_OUT ve TRANSFER tipinde olmayan tüm satırların düşürülmesiyle oluşturulmuştur. Bu veri setinde ise toplamda 2770409 satır veri bulunmaktadır ve 8213 tanesi fraud olan işlemdir.

6.1 K Nearest Neighbour

Bulunan en iyi parametreler n_neighbors 1 ve p 2 şeklindedir.

	Fit Time	Test Time	Accuracy	Recall	Precision	F1-Score	AUC
Undersampling	0.02	0.25	0.99	0.99	0.98	0.99	0.99
Tüm	4.89	78.37	1.0	0.84	0.78	0.81	0.92

6.2 Naive Bayes

Bulunan en iyi parametreler var_smoothing 3.5111917342151275e-06 şeklindedir.

	Fit Time	Test Time	Accuracy	Recall	Precision	F1-Score	AUC
Undersampling	0,0006	0,0010	0,69	1,00	0,62	0,76	0,61
Tüm	0,45	0,05	0,98	0,56	0,09	0,16	0,89

6.3 Logistic Regression

Bulunan en iyi parametreler C 100, penalty l2 ve solver liblinear şeklindedir.

	Fit Time	Test Time	Accuracy	Recall	Precision	F1-Score	AUC
Undersampling	0,0950	0,0020	0,93	0,93	0,93	0,93	0,98
Tüm	14,4900	0,1820	1,00	0,39	0,85	0,54	0,97

6.4 Decision Tree

Bulunan en iyi parametreler criterion gini, max_depth 10 ve min_samples_leaf 5 şeklindedir.

	Fit Time	Test Time	Accuracy	Recall	Precision	F1-Score	AUC
Undersampling	0,0840	0,0050	0,93	0,99	0,99	0,99	0,94
Tüm	12,1720	0,0450	1,00	0,61	0,89	0,73	0,99

6.5 Random Forest

En iyi parametreler max_depth 20, min_samples_leaf ve 5, n_estimators 200 şeklindedir.

	Fit Time	Test Time	Accuracy	Recall	Precision	F1-Score	AUC
Undersampling	7,2780	0,0970	0,99	1,00	0,99	0,99	1,00
Tüm	1.207,0400	10,4200	1,00	0,79	0,93	0,85	0,99

6.6 Gradient Boosting

En iyi parametreler max_depth 8, min_samples_leaf ve 4, n_estimators 100 şeklindedir.

	Fit Time	Test Time	Accuracy	Recall	Precision	F1-Score	AUC
Undersampling	4,5110	0,0180	0,99	1,00	0,99	0,99	1,00
Tüm	885,5200	1,3300	1,00	0,57	0,85	0,68	0,99

6.7 Sonuçların Karşılaştırılması

Undersampling yapılarak oluşturulan modeller için başarı oranı Naive Bayes ve Logistic Regression hariç 0.99 iken, Fraud olmayan tipteki işlemleri silerek oluşturduğumuz veri setinde modellerin başarı oranları daha düşük kalmıştır. Random Forest 0.85 başarı oranıyla en iyi model olmuştur. Fakat eğitim süresi 1207 saniye ile çok uzundur. Buna karşılık 0.81 başarı oranı ve 11 saniyelik eğitim süresiyle kNN modeli de başarılıdır.

7. Kaynakça

[1] Kazım SOYLU, Kredi Kartı Sahte İşlem Tespiti, 2022
[2] Ali Kemal AY, Kredi Kartı Dolandırıcılığının Tespitinde Yeniden Örnekleme Tekniklerinin Kullanımı, 2022
[3] Hamzah Ali Shukaur Almarsoomi, Credit Card Fraud Detection Using Machine Learning Methodology, 2019
[4] Muhammad Qasim Raza, Credit Card Detection Using Machine Learning Algorithm And Analysis Based on Time Series Data