Kredi Kartı Dolandırıcılığı Tespiti: Gerçekçi Bir Modelleme ve Yeni Bir Öğrenme Stratejisi

Andrea Dal Pozzolo, Giacomo Boracchi, Olivier Caelen, Cesare Alippi, *Fellow, IEEE,* ve Gianluca Bontempi, *Senior Member, IEEE*

Özet- Kredi kartı işlemlerindeki sahtekarlıkları tespit etmek, hesaplamalı zeka algoritmaları için belki de en iyi test alanlarından biridir. Aslında bu problem, kavram kayması (müşterilerin alışkanlıkları gelişir ve dolandırıcılar zaman içinde stratejilerini değiştirir), sınıf dengesizliği (gerçek işlemler sahtekarlıklardan çok daha fazladır) ve doğrulama gecikmesi (yalnızca küçük bir işlem kümesi araştırmacılar tarafından zamanında kontrol edilir) gibi bir dizi ilgili zorluğu içerir. Bununla birlikte, dolandırıcılık tespiti için önerilen öğrenme algoritmalarının büyük çoğunluğu, gerçek dünyadaki bir dolandırıcılık tespit sisteminde (FDS) neredeyse hiç geçerli olmayan varsayımlara dayanmaktadır. Bu gerçekçilik eksikliği iki ana hususla ilgilidir: 1) denetimli bilginin sağlanma şekli ve zamanlaması ve
2) dolandırıcılık tespit performansını değerlendirmek için kullanılan ölçütler. Bu makalenin üç önemli katkısı bulunmaktadır. İlk olarak, endüstriyel ortağımızın yardımıyla, her gün kredi kartı işlemlerinin büyük akışlarını analiz eden FDS'lerin çalışma koşullarını gerçekçi bir şekilde tanımlayan dolandırıcılık tespit probleminin bir formalizasyonunu öneriyoruz. Ayrıca dolandırıcılık tespiti amacıyla kullanılabilecek en uygun performans ölçütlerini de gösteriyoruz. İkinci olarak, sınıf dengesizliğini, kavram kaymasını ve doğrulama gecikmesini etkili bir şekilde ele alan yeni bir öğrenme stratejisi tasarlıyor ve değerlendiriyoruz. Üçüncü olarak, deneylerimizde, üç yıllık bir zaman aralığında yetkilendirilmiş 75 milyondan fazla işlem içeren gerçek dünya veri akışında sınıf dengesizliği ve kavram kaymasının etkisini gösteriyoruz.

***Dizin* Terimleri- Kavram kayması, Kredi kartı dolandırıcılığı tespiti, durağan olmayan ortamlarda öğrenme, dengesiz sınıflandırma.**

I. Giriş

CREDIT kart dolandırıcılığı tespiti, makine öğrenimi ve bilgisayarlı istihbarat topluluklarının dikkatini çeken ve çok sayıda

Makale 8 Aralık 2015 tarihinde alındı; 26 Ağustos 2016 tarihinde revize edildi ve
28 Şubat 2017; kabul tarihi 24 Temmuz 2017. Yayın tarihi 14 Eylül 2017; güncel versiyon tarihi 18 Temmuz 2018. A. Dal Pozzolo'nun çalışması *Doctiris* Bursu tarafından desteklenmiştir. G. Bontempi'nin çalışması Belçika Innoviris tarafından finanse edilen *BridgeIRIS* ve *BruFence tarafından* desteklenmiştir. *(Sorumlu yazar: Andrea Dal Pozzolo.)*

A. Dal Pozzolo ve G. Bontempi, Université Libre de Bruxelles, Bilgisayar Bilimleri Bölümü, Makine Öğrenimi Grubu, 1050 Brüksel, Belçika'da çalışmaktadır (e-posta: adalpozz@ulb.ac.be; gbonte@ulb.ac.be).

G. Boracchi Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, İtalya'da çalışmaktadır (e-posta: giacomo.boracchi@polimi.it).

O. Caelen, Worldline, 1130 Brüksel, Belçika, Ar-Ge Yüksek İşleme ve Hacim Ekibinde çalışmaktadır (e-posta: olivier.caelen@worldline.com).

C. Alippi, Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, İtalya ve ayrıca Università della Svizzera Italiana, 6900 Lugano, İsviçre'de çalışmaktadır (e-posta: cesare.alippi@polimi.it).

Bu makalede yer alan şekillerden bir veya daha fazlasının renkli versiyonlarına http://ieeexplore.ieee.org adresinden ulaşabilirsiniz.
Dijital Nesne Tanımlayıcı 10.1109/TNNLS.2017.2736643

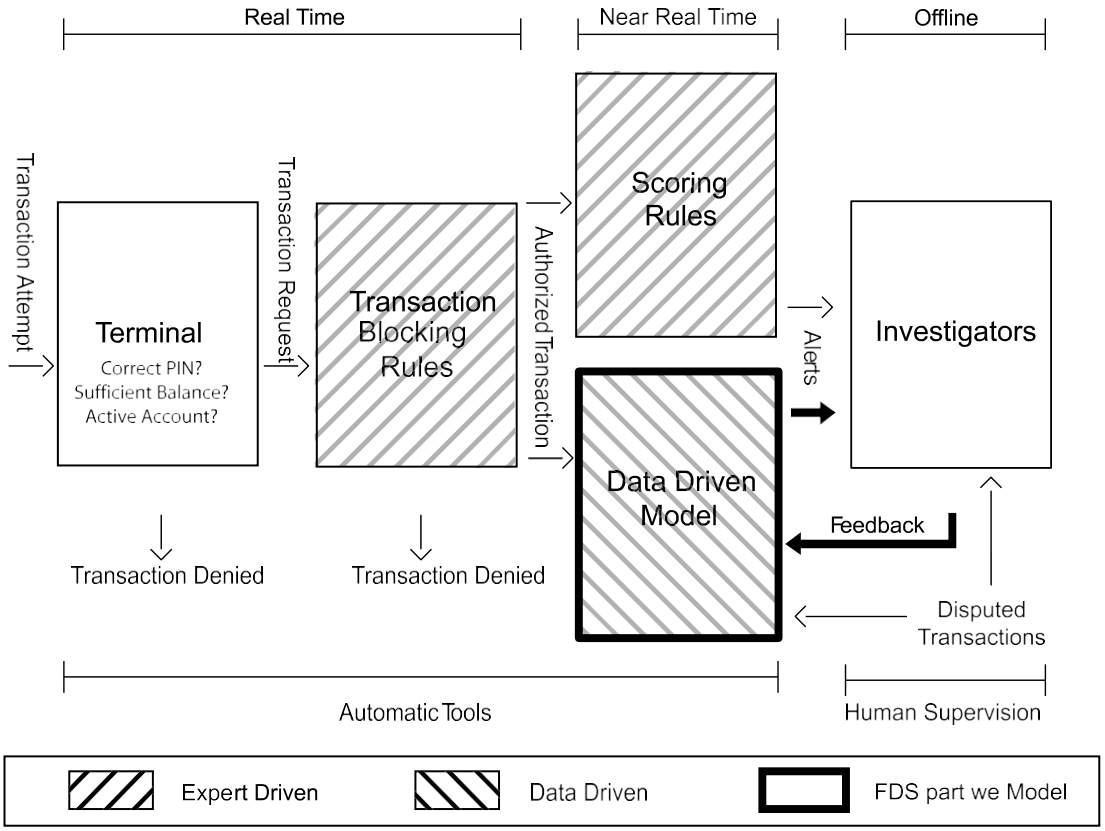
otomatik çözümler önerilmiştir [1], [6], [8], [23], [24], [41], [47], [55], [56], [66]. Aslında, bu problem öğrenme açısından özellikle zorlayıcı görünmektedir, çünkü aynı zamanda *sınıf dengesizliği* [21], [22], yani gerçek işlemlerin sahtekarlıklardan çok daha fazla olması ve *kavram kayması [*4], [35], yani işlemlerin zaman içinde istatistiksel özelliklerini değiştirebilmesi ile karakterize edilmektedir. Ancak bunlar, gerçek dünyadaki bir sahtekarlık tespit sistemindeki (FDS) öğrenme sorunlarını karakterize eden tek zorluklar değildir.

Gerçek dünyadaki bir FDS'de, devasa ödeme talebi akışı, hangi işlemlerin yetkilendirileceğini belirleyen otomatik araçlar tarafından hızla taranır. Sınıflandırıcılar genellikle tüm yetkilendirilmiş işlemleri analiz etmek ve en şüpheli olanları *uyarmak için kullanılır*. Uyarılar daha sonra, uyarılan her işlemin gerçek niteliğini (gerçek veya hileli) belirlemek için kart sahipleriyle iletişime geçen profesyonel araştırmacılar tarafından incelenir. Bunu yaparak araştırmacılar, zaman içinde dolandırıcılık tespit performansını korumak (veya nihayetinde iyileştirmek) amacıyla sınıflandırıcıyı eğitmek veya güncellemek için kullanılabilecek etiketli işlemler şeklinde sisteme bir *geri bildirim* sağlar. İşlemlerin büyük çoğunluğu, bariz zaman ve maliyet kısıtlamaları nedeniyle müfettişler tarafından doğrulanamaz. Bu işlemler, müşteriler sahtekarlıkları keşfedip bildirene kadar veya tartışmalı olmayan işlemlerin gerçek kabul edileceği yeterli bir süre geçene kadar etiketsiz kalır.

Bu nedenle, uygulamada, denetlenen örneklerin çoğu, *doğrulama gecikmesi olarak* bilinen bir sorun olan önemli bir gecikmeyle sağlanır [44]. Sınıflandırıcıyı güncellemek için mevcut olan tek yeni denetimli bilgi, uyarı-geri *bildirim etkileşimi* yoluyla sağlanır. Literatürdeki çoğu makale doğrulama gecikmesini [53] ve uyarı-geri bildirim etkileşimini göz ardı etmekte ve gerçekçi olmayan bir şekilde her bir işlemin etiketinin düzenli olarak, örneğin günlük olarak FDS'ye sunulduğunu varsaymaktadır (bkz. [6], [8], [12], [23], [24], [28], [47], [55]).

Bununla birlikte, kavram kayması meydana geldiğinde doğrulama gecikmesi zararlı olduğundan ve uyarı-geribildirim etkileşimi, eğitim ve test verilerinin dağılımı arasında daha fazla fark yaratan bir tür *örnek seçim yanlılığından (*SSB) [19] sorumlu olduğundan, gerçek dünyada bir FDS tasarlanırken bu hususlar dikkate alınmalıdır.

Literatürde tipik olarak yapılanlar ile Sahtekarlık Tespit Sisteminin (FDS) gerçek dünyadaki çalışma koşulları arasındaki bir diğer önemli fark, sahtekarlık tespit performansını değerlendirmek için kullanılan ölçütlerle ilgilidir. Çoğu zaman, küresel sıralama ölçütleri [23], [24], [63], örneğin



Şekil 1. Bir FDS'deki kontrol katmanlarını gösteren şema. Odak noktamız esas olarak DDM ve son denetimli örneklerin sağlanma şeklini düzenleyen *uyarı-geri bildirim* etkileşimidir.

ROC eğrisi (AUC) veya maliyet tabanlı ölçütler [6], [47], [55] kullanılmaktadır, ancak bunlar her gün yalnızca birkaç uyarının kontrol edilebileceği ve şirketlerin üretilen uyarıların hassasiyeti konusunda çok endişeli olduğu gerçeğini göz ardı etmektedir.

Bu makalenin temel katkıları aşağıdaki gibidir.

1) Gerçek dünyadaki bir FDS'yi düzenleyen mekanizmaları tanımlıyor ve dolandırıcılık tespitinde ele alınması gereken eklemli sınıflandırma probleminin resmi bir modelini sunuyoruz.
2) Gerçek dünyadaki bir FDS'de dikkate alınan performans ölçütlerini tanıtıyoruz.
3) Bu sağlam ve gerçekçi model dahilinde, doğrulama gecikmesi ve uyarı-geri bildirim etkileşimi de dahil olmak üzere yukarıdaki zorlukları ele almak için etkili bir öğrenme stratejisi öneriyoruz. Bu öğrenme stratejisi çok sayıda kredi kartı işlemi üzerinde test edilmiştir.

Bu makale aşağıdaki şekilde düzenlenmiştir. İlk olarak Bölüm II'de gerçek dünyadaki bir FDS'nin çalışma koşullarını detaylandırıyoruz ve ardından Bölüm III'te ifade edilen dolandırıcılık tespit problemini modelliyor ve en uygun performans ölçütlerini sunuyoruz. Özellikle, tespit edilen hileli işlemlerin (veya kartların) sayısını, araştırmacıların kontrol edebileceği maksimum işlem (veya kart) sayısı üzerinden değerlendirmenin en uygun yöntem olduğunu düşünüyoruz. Daha sonra Bölüm IV'te dolandırıcılık tespiti amacıyla bir sınıflandırıcı eğitilirken karşılaşılan temel zorluklar ele alınmaktadır. Bölüm V, geri bildirimlerden ve gecikmeli denetimli örneklerden farklı sınıflandırıcıları ayrı ayrı eğitmeyi ve ardından tahminlerini toplamayı içeren önerilen öğrenme stratejisini tanıtmaktadır. Bu strateji, geri bildirimlerden ve

Geri bildirimlerin ve gecikmeli denetimli örneklerin farklı doğasının, kayan pencere veya sınıflandırıcı topluluğu kullanan FDS'de özellikle etkili olduğu gösterilmiştir. İddialarımızı, üç yıl boyunca elde edilen 75 milyondan fazla e-ticaret kredi kartı işlemi üzerinde yapılan deneylerde (Bölüm VI) doğrulamaktayız; bu işlemler ayrıca gerçek dünyadaki işlem akışlarında sınıf dengesizliği ve kavram kaymasının etkisini gözlemlemek için analiz edilmektedir.

Çalışmamız, bir FDS'nin gerçek dünyadaki çalışma koşullarını ayrıntılı olarak tanımlayarak ve uyarı-geri bildirim etkileşiminin getirdiği SSB'yi analiz ederek önemli ölçüde genişletilmiş olan [20] üzerine inşa edilmiştir. Ayrıca, deneysel bölüm büyük ölçüde güncellenmiş ve iki büyük veri seti üzerinde ek analizler sunularak tamamlanmıştır.

II. REAL-WORLD FDS

Burada, endüstriyel ortağımız tarafından rutin olarak kullanılan bir FDS'den esinlenerek gerçek dünyadaki bir FDS'nin temel özelliklerini ve çalışma koşullarını açıklıyoruz. Şekil 1'de tipik olarak bir F D S ' d e kullanılan beş kontrol katmanı gösterilmektedir: 1) terminal;
2) işlem engelleme kuralları; 3) puanlama kuralları; 4) veri odaklı model (DDM); ve 5) araştırmacılar.

Katman 1)-4) otomatik kontrolleri tam olarak uygularken, katman 5) insan müdahalesi gerektiren tek katmandır.

A. Bir FDS'deki Kontrol Katmanları

1) Terminal: Terminal, bir FDS'deki ilk kontrol katmanını temsil eder ve tüm ödeme talepleri üzerinde geleneksel güvenlik kontrolleri gerçekleştirir [63]. Güvenlik kontrolleri şunları içerir

PIN kodu (sadece çipli kartlarda mümkündür), deneme sayısı, kart durumu (aktif veya bloke), mevcut bakiye ve harcama limiti. Çevrimiçi işlemlerde, bu işlemlerin gerçek zamanlı olarak gerçekleştirilmesi gerekir (yanıt birkaç milisaniye içinde sağlanmalıdır), bu sırada terminal kartı veren şirketin bir sunucusunu sorgular. Bu kontrollerden herhangi birini geçemeyen talepler reddedilirken, diğerleri ikinci kontrol katmanı tarafından işlenen işlem talepleri haline gelir.

2) İşlem Engelleme Kuralları: İşlem engelleme kuralları, açıkça dolandırıcılık olarak algılanan işlem taleplerini engellemeyi amaçlayan *if-then (-else)* ifadeleridir. Bu kurallar, geçmiş kayıtları veya kart sahibi profilini analiz etmeden, ödeme talep edildiğinde mevcut olan az sayıdaki bilgiyi kullanır. Engelleme kuralına örnek olarak "*EĞER İNTERNET İŞLEMLERİ VE GÜVENLİ OLMAYAN WEB SİTESİ O ZAMAN İŞLEMİ REDDET*" verilebilir.[1] Uygulamada, birkaç işlem engelleme kuralı eş zamanlı olarak yürütülür ve bu kurallardan herhangi birini ihlal eden işlemler engellenir (ancak kartlar devre dışı bırakılmaz). İşlem engelleme kuralları araştırmacı tarafından manuel olarak tasarlanır ve bu nedenle FDS'nin *uzman güdümlü* bileşenleridir. Gerçek zamanlı işlemleri güvence altına almak ve birçok gerçek işlemi engellemekten kaçınmak için engelleme kuralları aşağıdaki özelliklere sahip olmalıdır 1) hızlı hesaplanmalı ve 2) çok hassas, yani çok az yanlış alarm vermelidir. Engelleme kurallarını geçen tüm işlemler nihayet rized. Bununla birlikte, dolandırıcılık tespit faaliyeti, mevcut satın alma işlemini öncekilerle ve kart sahibi profiliyle karşılaştırmak için kullanılan toplu özelliklerle işlem verilerini zenginleştirdikten sonra devam eder. Bu birleştirilmiş özellikler arasında örneğin ortalama harcama, aynı gün içindeki ortalama işlem sayısı veya önceki alışverişlerin yeri yer alır. Birleştirilmiş özelliklerin hesaplanması süreci *özellik artırımı* olarak adlandırılır ve Bölüm II-B'de açıklanmıştır. Artırılmış özellikler ve mevcut işlem verileri, yetkilendirilmiş işlemin hileli mi yoksa gerçek mi olduğunu belirlemek için bilgilendirici olması beklenen bir özellik *vektöründe* istiflenir. FDS'nin aşağıdaki katmanları bu özellik vektörü üzerinde çalışır.

3) Puanlama Kuralları: Puanlama kuralları da *if-then (-else)* ifadeleri olarak ifade edilen uzman güdümlü modlardır. Ancak bunlar özellik vektörleri üzerinde çalışır ve her yetkili işleme bir puan atar: puan ne kadar büyükse, işlemin dolandırıcılık olma olasılığı o kadar yüksektir. Puanlama kuralları müfettişler tarafından manuel olarak tasarlanır ve ilgili puanları keyfi olarak tanımlar. Puanlama kuralına örnek olarak "*EĞER önceki işlem farklı bir kıtada VE önceki işlemden 1 saatten daha kısa bir süre sonra gerçekleşmişse O ZAMAN dolandırıcılık puanı = 0,95*" verilebilir.

Ne yazık ki, puanlama kuralları yalnızca araştırmacılar tarafından daha önce keşfedilmiş olan ve özellik vektörlerinin birkaç bileşenini içeren kalıplar sergileyen hileli stratejileri tespit edebilir. Dahası, farklı uzmanlar farklı kurallar tasarladığı için puanlama kuralları oldukça özneldir.

4) Veri Güdümlü Model (DDM): Bu katman tamamen veri odaklıdır ve her bir özellik vektörünün dolandırıcılık olma olasılığını tahmin etmek için bir sınıflandırıcı veya başka bir istatistiksel model kullanır. Bu olasılık, yetkilendirilmiş işlemlerle ilişkili dolandırıcılık puanı olarak kullanılır. Böylece, DDM bir dizi etiketli işlemden eğitilir ve araştırmacılar tarafından yorumlanamaz veya manuel olarak değiştirilemez. Etkili bir DDM'nin, özellik vektörünün çoklu bileşenlerini eş zamanlı olarak analiz ederek, muhtemelen kulak dışı ifadeler yoluyla hileli örüntüleri tespit etmesi beklenir. Bu nedenle, DDM'nin müfettiş deneyiminin ötesine geçen ve yorumlanabilir kurallara karşılık gelmesi gerekmeyen kurallara göre sahtekarlıkları bulması beklenir.

Bu makale, FDS'nin bu bileşenine odaklanmakta ve dolandırıcılık tespit performansını artırmak için DDM'yi tasarlamak, eğitmek ve güncellemek için bir strateji önermektedir. Büyük bir dolandırıcılık puanı alan ya da dolandırıcılık olma olasılığı yüksek olan özellik vektörleriyle ilişkili işlemler uyarı oluşturur. Yalnızca sınırlı sayıda uyarılmış işlem, son kontrol katmanını temsil eden müfettişlere bildirilir.

5) Araştırmacılar: Araştırmacılar, kredi kartı işlemlerini analiz etme konusunda deneyimli profesyonellerdir ve FDS'nin uzman güdümlü katmanlarından sorumludurlar. Araştırmacılar özellikle işlem engelleme ve puanlama kurallarını tasarlarlar.

Müfettişler ayrıca puanlama kuralları ve DDM tarafından oluşturulan uyarıları kontrol ederek bunların dolandırıcılık mı yoksa yanlış alarm mı olduğunu belirlemekle görevlidir. Özellikle, uyarılan tüm işlemleri, uygulamada her bir işlemin ne kadar *riskli* olduğunu gösteren atanmış puanlar/olasılıklar da dahil olmak üzere işlemle ilgili tüm bilgilerin raporlandığı bir vaka yönetim aracında görselleştirirler. Müfettişler kart sahiplerini arar ve doğruladıktan sonra uyarılan işleme "gerçek" veya "hileli" etiketini atar ve bu bilgiyi FDS'ye geri gönderir. Aşağıda, bu etiketli işlemlerden *geri bildirim* olarak bahsedeceğiz ve gerçek dünyadaki bir FDS'de denetimli bilgi sağlayan bu mekanizmayı tanımlamak için *uyarı-geri bildirim etkileşimi* terimini kullanacağız.

Dolandırıcılık kurbanı olduğu tespit edilen herhangi bir kart, daha fazla dolandırıcılık faaliyetini önlemek için derhal bloke edilir. Tipik olarak, müfettişler taahhüt edilen bir karttan yapılan tüm son işlemleri kontrol eder, bu da tespit edilen her dolandırıcılığın potansiyel olarak birden fazla geri bildirim oluşturabileceği anlamına gelir, ancak bunların uyarılara veya dolandırıcılıklara karşılık gelmesi gerekmez. Gerçek dünyadaki bir FDS'de, araştırmacılar günde yalnızca birkaç uyarıyı kontrol edebilir [45] çünkü bu süreç uzun ve sıkıcı olabilir. Bu nedenle, bir DDM'nin birincil hedefi kesin uyarılar vermektir, çünkü çok fazla yanlış alarm rapor edildiğinde müfettişler diğer uyarıları göz ardı edebilir.

B. Özellikler Büyütme

Herhangi bir işlem talebi, satıcı kimliği, kart sahibi kimliği, satın alma tutarı, tarih ve saat gibi birkaç değişkenle tanımlanır. Engelleme kurallarını geçen tüm işlem talepleri, özellik artırma sürecinin başladığı tüm son yetkili işlemleri içeren bir veritabanına girilir. Özellik artırımı sırasında, satın alma hakkında ek bilgi sağlamak ve sahtekarlıkları gerçek işlemlerden daha iyi ayırt etmek için her yetkili işlemle ilişkili belirli bir *toplu özellik* kümesi hesaplanır. Birleştirilmiş özelliklere örnek olarak müşterinin her hafta/ay yaptığı ortalama harcama, günlük veya aynı mağazadaki ortalama

[1]Bu kurallar gizlidir ve hiçbirini ifşa edemeyiz. Burada, bu kurallarda ne tür bilgilerin kullanılabileceğini göstermek için gerçekçi bir örnek sunuyoruz.

işlem sayısı, ortalama işlem tutarı ve son alışverişlerin yapıldığı yer verilebilir [7], [8], [23], [41],

[45], [66]. Van Vlasselaer *ve diğerleri* [63] kart sahiplerini tüccarlara/dükkanlara bağlayan sosyal ağlardan ek bilgilendirici özelliklerin çıkarılabileceğini göstermiştir.

Toplu özellikler, kart sahibinin son faaliyetlerini özetledikleri için çok bilgilendiricidir. Böylece, kendi başlarına şüpheli olmayan ancak belirli kart sahibinin alışveriş alışkanlıklarına kıyasla olağandışı olabilecek işlemlerin uyarılmasına olanak tanırlar. Özelliklerin artırılması hesaplama açısından pahalı olabilir ve birleştirilmiş özellikler genellikle geçmiş işlemler temelinde her kart sahibi için çevrimdışı olarak önceden hesaplanır. Toplanan özellikler, özellik vektöründeki işlem verileriyle birlikte istiflenir.

C. Denetimli Bilgi

Müfettişlerin geri bildirimleri FDS'ye sunulan en son denetimli bilgilerdir, ancak her gün işlenen işlemlerin yalnızca küçük bir bölümünü temsil eder [20]. Ek etiketli işlemler, yetkisiz işlemlere doğrudan itiraz eden kart sahipleri tarafından sağlanır [20], [63]. Kart sahipleri banka tarafından gönderilen kredi kartı dökümünü kontrol ederken farklı alışkanlıklara sahip olduğundan, itiraz edilen işlemlerin zamanlaması önemli ölçüde değişebilir. Ayrıca, ihtilaflı işlemlerin kontrol edilmesi, önemli gecikmelere neden olabilecek bazı gerekli idari prosedürleri gerektirir.

Diğer tüm işlemler etiketsiz kalır: bunlar ya gerçek işlemler ya da FDS tarafından gözden kaçırılan ve kart sahipleri tarafından göz ardı edilen sahtekarlıklar olabilir. Ancak, kart sahibinin itirazı olmadan belirli sayıda gün geçtikten sonra, bildirilmeyen tüm işlemler varsayılan olarak gerçek kabul edilir ve DDM'nin eğitim setine eklenir.

Genel olarak, iki tür denetimli bilgi vardır:
1) müfettişler tarafından sağlanan ve sayıları sınırlı olan ancak son işlemlere atıfta bulunan geri bildirimler ve 2) büyük çoğunluğu birkaç gün sonra (örneğin bir ay) etiketlere ulaşılabilen gecikmeli denetimli işlemler. Bu sonuncusu hem ihtilaflı hem de ihtilafsız işlemleri içerir.

D. Sistem Güncellemesi

Müşterilerin harcama davranışları gelişmekte ve dolandırıcılar sürekli olarak yeni saldırılar tasarlamakta, dolayısıyla stratejileri de zaman içinde değişmektedir. Bu durumda tatmin edici bir performansı garanti etmek için FDS'yi sürekli güncellemek gerekir. Uzman güdümlü sistemler, yeni dolandırıcılık faaliyetlerinin başlangıcına karşı koymak için geçici (işlem engelleme veya puanlama) kurallar ekleyen ve çok fazla yanlış uyarıdan sorumlu olan kuralları kaldıran araştırmacılar tarafından düzenli olarak güncellenir. Ancak, araştırmacılar DDM'yi değiştiremez, çünkü yorumlanabilir değildir ve Şekil 1'de gösterildiği gibi yalnızca son denetimli bilgiler temelinde güncellenebilir (örneğin, yeniden eğitilebilir). Bu işlem tipik olarak çok sayıda etiketli işlem gerektirir; bu nedenle, araştırmacılar gün boyunca sürekli olarak geri bildirim sağlasa da, sınıflandırıcı genellikle yalnızca bir kez, özellikle de yeterli sayıda geri bildirim mevcut olduğunda gün sonunda güncellenir / yeniden eğitilir.

III. Problem Formülasyonu

Burada, gerçek dünyadaki bir FDS'de ele alınması gereken sınıflandırma problemini modelleyerek, FDS'nin resmi bir tanımını sağlıyoruz.

uyarı-geribildirim etkileşimi ve uygun performans ölçütlerinin sunulması. Önerilen öğrenme stratejisi (Bölüm V) ve deneylerimiz (Bölüm VI) bu model üzerine inşa edilmiştir.

x_i i'inci yetkili işlemle ilişkili özellik vektörünü ve $y_i \in \{+, -\}$ ilgili sınıfı göstersin; burada + dolandırıcılığı ve - gerçek bir işlemi ifade eder. İşlem akışının zamanla değişen doğasıyla başa çıkmak için,

K sınıflandırıcısı her gün güncellenir (veya yeniden eğitilir). Özellikle, t -1 gününe kadar mevcut olan denetimli işlemler üzerinde eğitilen sınıflandırıcıyı K_{t-1} ile gösteriyoruz. K_{t-1} sınıflandırıcısı daha sonra t gününde yetkilendirilmiş T_t işlem kümesini işlemek için kullanılır. $_{PKt-1}(+|x_i)$ ile K_{t-1} 'nin ardılını, yani x_i 'nin K_{t-1} 'ye göre dolandırıcılık olma olasılığını gösteriyoruz. Müfettişler yalnızca birkaç yüksek riskli işlemi kontrol eder. Bu nedenle, *uyarıları* k-en riskli işlem olarak modellemekteyiz.

işlemler, yani

$$A_t = \{x_i \in T_t \text{ s.t. } r(x_i) \leq k\} \quad (1)$$

burada $r(x_i) \in \{1, \ldots, |T_t|\}$ x_i 'in $_{PKt}(+|x_i)$'ye göre sıralaması ve $k > 0$ araştırmacılar tarafından kontrol edilebilecek maksimum uyarı sayısıdır.[2] Daha önce de belirtildiği gibi

Bölüm II-A.5, müfettişler kart sahipleriyle iletişime geçer ve FDS'ye *geri bildirim* şeklinde denetimli örnekler sunar. Özellikle, geri bildirimler, kontrol edilen kartlardan yapılan tüm son işlemleri içerir ve bunları şu şekilde modelleriz

$$F_t = \{(x_i, y_i) \text{ s.t. } x_i \text{ kartlardan}(A_t)\} \quad (2)$$

Burada kartlar(A_t), A_t 'da en az bir işlemi olan kartlar kümesini ifade eder. Geri bildirimlerin sayısı, yani $|F_t|$, k kontrollü kartla ilişkili işlemlerin sayısına bağlıdır.

Belirli bir doğrulama gecikmesinden sonra, tüm işlemlerin etiketleri FDS'ye sağlanır, çünkü Bölüm II-C'de tartışıldığı gibi, tartışmalı olmayan işlemler gerçek kabul edilir. Basitlik adına, δ günlük sabit bir doğrulama gecikmesi varsayıyoruz, öyle ki t gününde tüm işlemlerin etiketleri

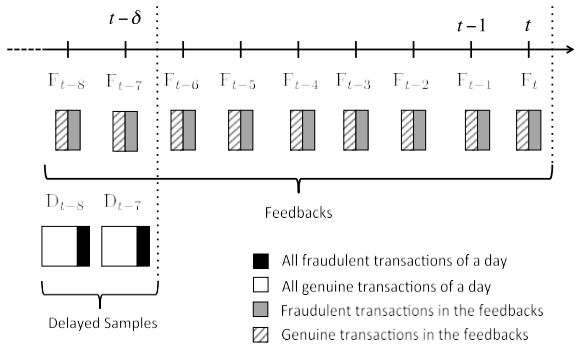
$t - \delta$ gününde yetkilendirilen işlemler sağlanır. Bunları *gecikmeli denetimli örnekler* olarak adlandırıyoruz

$$D_{t-\delta} = \{(x_i, y_i), x_i \in T_{t-\delta}\}. \quad (3)$$

$F_{t-\delta} \subset D_{t-\delta}$ olduğunu unutmayın, çünkü t -δ günündeki işlemler açıkça uyarılmış olanları içerir. Şekil 2, bir FDS'de mevcut olan farklı denetimli bilgi türlerini göstermektedir.

Resmi tanımımıza rağmen şunu belirtmekte fayda var dolandırıcılık tespit literatüründe şimdiye kadar göz ardı edilen çeşitli yönleri ve ayrıntıları içermekle birlikte, bu hala basitleştirilmiş bir modeldir. Aslında, gerçek dünyadaki bir FDS'deki uyarılar, T_t adresindeki tüm işlemleri sıralamak zorunda kalmadan, işlemler işlenirken genellikle çevrimiçi olarak verilir. Benzer şekilde, her bir ihtilaflı işlem δ günden daha az (veya muhtemelen daha fazla) sürebileceğinden, gecikmeli denetimli çiftler bir kerede gelmez. Bununla birlikte, formülasyonumuzun gerçek dünyadaki bir FDS'nin öğrenme perspektifinden en önemli olan yönlerini dikkate aldığını düşünüyoruz; bunlar arasında uyarılar, değiştirme-geri bildirim etkileşimi ve doğrulama gecikmesi yer almaktadır. Ayrıca, prensip olarak, sınıflandırıcının

[2]Bu makale $^{\text{boyunca,}}$ bir kümenin kardinalitesini |-| ile göstereceğiz.



Şekil 2. *t* gününün sonunda mevcut olan denetimli örnekler şunları içerir: 1) *t* - δ gününden önce meydana gelen geri bildirimler [$F_{(-)}$] ve 2) gecikmeli çiftler [$D_{(-)}$]. Bu grafikte δ = 7 olarak varsayılmıştır. Desenler farklı etiketleri gösterir ve bu bölgelerin boyutu dengeli/dengesiz sınıf oranlarını gösterir.

her bir özellik vektörünü x_i bağımsız olarak analiz eder, bunlardan herhangi biri uyarı havuzuna girene kadar birkaç riskli işlem alan kartları uyarmaz (1). Bununla birlikte, bu durumlar özellikle müfettişler için önemlidir ve uygun puanlama kuralları ya da özellik artırımı, örneğin son işlemlerin puanlarını takip eden bir bileşen eklenerek e l e alınabilir.

Dolandırıcılık tespit performansı, aşağıdaki gibi tanımlanan P_k *(t) uyarı hassasiyeti* açısından uygun bir şekilde değerlendirilebilir

$$P_k(t) = \frac{|TP_k(t)|}{k} \quad (4)$$

burada TP_k *(t)* = {(x_i , y_i) öyle ki $x_i \in A_t$, y_i = +}. Böylece, P_k *(t),* A_t uyarılarındaki sahtekarlıkların oranıdır. Yine de
sınıflandırıcı her bir özellik vektörünü bağımsız olarak işlediğinden, uyarı hassasiyeti yetkili işlemler yerine kartlar açısından daha gerçekçi bir şekilde ölçülebilir. Aslında, A_t adresinde aynı karttan yapılan birden fazla işlem tek bir uyarı olarak sayılmalıdır, çünkü müfettişler kart sahipleriyle iletişime geçerken tüm son işlemleri kontrol eder. Bu da *k'nın* müfettişlerin kontrol edebileceği maksimum kart sayısına bağlı olduğu anlamına gelir. Bu bağlamda, tespit performansını kart düzeyinde ölçmek daha bilgilendiricidir, öyle ki aynı karttan yapılan birden fazla hileli işlem tek bir doğru tespit olarak sayılır. Bu nedenle, CP_k , *kart hassasiyetini,* müfettişler tarafından kontrol edilen *k kartta* tespit edilen hileli kartların oranı olarak tanıtıyoruz

$$CP_k(t) = \frac{|C_t|^+}{k} \quad (5)$$

Burada $_tC^+$ *t* gününde doğru tespit edilen hileli kartlar kümesini, yani en az bir uyarı bildiren hileli kartları ifade etmektedir. *k'dan* daha az kartın hileli olduğu günleri doğru bir şekilde hesaba katmak için *normalleştirilmiş* CP_k *(t)'yi* şu şekilde tanımlarız

$$NCP_k(t) = \frac{CP_k(t)}{\Gamma(t)} \text{ ile } \Gamma(t) = \begin{cases} 1 & \text{eğer } \gamma_t \geq k \text{ ise} \\ \frac{\gamma_t}{k} & \text{aksi halde} \end{cases} \quad (6)$$

t gününde, müfettişler tarafından kontrol edilen *k* = 100 karttan 40 sahte kartı doğru bir şekilde tespit etmişsek ($|C_t^+|$ = 40) ve toplam sahte kart sayısı 50 ise (γ_t = 50), CP_k *(t)* = 0,4 ve NCP_k *(t)* = *(0,4)/(0,5)* = 0,8 olur.

Γ(t) benimsenen belirli K_{t-1} sınıflandırmasına bağlı olmadığından, "A" algoritması CP_k açısından "B" algoritmasından daha iyi olduğunda, "A "nın NCP_k açısından da "B "den daha iyi olduğunu unutmayın. Ayrıca, doğrulama gecikmesi nedeniyle, *t* günündeki hileli kart sayısı (yani γ_t) ancak birkaç gün sonra hesaplanabilir ve bu nedenle NCP_k
gerçek zamanlı. Bu nedenle, çalışma performansını değerlendirmek için CP_k , Bölüm VI-F'de olduğu gibi farklı FDS konfigürasyonlarını test ederken geriye dönük test için NCP_k kullanmanızı öneririz.

IV. İLGILI ÇALIŞMALAR

A. Kredi Kartı Dolandırıcılığının Tespitinde Veriye Dayalı Yaklaşımlar

Kredi kartı dolandırıcılığı tespiti için hem denetimli [8], [12], [15] hem de denetimsiz [11], [14], [62] yöntemler önerilmiştir. Denetimsiz yöntemler, çoğunluğa uymayan herhangi bir işlemi dolandırıcılık olarak değerlendiren aykırı değer / anormallik tespit tekniklerinden oluşur. Dikkat edilirse, bir FDS'deki denetimsiz bir DDM doğrudan etiketsiz işlemlerden elde edilebilir. İyi bilinen bir yöntem olan eş grup analizi [65], müşterileri profillerine göre kümelemekte ve sahtekarlıkları çoğunluktan ayrılan işlemler olarak tanımlamaktadır.
tipik kart sahibinin davranışından farklıdır (ayrıca bkz. [52]). Bu
tipik kart sahibinin davranışı da kendi kendini organize eden haritalar aracılığıyla modellenmiştir [51], [54], [71].

Denetimli yöntemler, dolandırıcılık tespitinde açık ara en popüler yöntemlerdir ve bir sınıflandırıcıyı eğitmek için etiketli işlemlerden yararlanırlar. Dolandırıcılıklar, yetkili işlemlerin özellik vektörleri sınıflandırılarak veya muhtemelen sınıflandırıcının son hali analiz edilerek tespit edilir [10]. Sahtekarlıkları tespit etmek için kredi kartı işlemleri üzerinde sinir ağları [1], [12], [28], lojistik regresyonlar da dahil olmak üzere çeşitli sınıflandırma algoritmaları test edilmiştir.
sion [41], birliktelik kuralları [56], destek vektör makineleri [66], değiştirilmiş Fisher diskriminant analizi [47] ve karar ağaçları [6], [24], [55]. Birçok çalışma rastgele ormanın (RF) en iyi performansı elde ettiğini bildirmiştir [8], [20], [23], [63], [66]: deneylerimizde RF'leri benimsememizin nedenlerinden biri budur.

B. Dolandırıcılık Tespiti için Performans Ölçütü

Dolandırıcılık tespiti için tipik performans ölçütü prob-
lemler AUC'dir [23], [24], [63]. AUC tahmin edilebilir
Mann-Whitney istatistiği aracılığıyla [48] ve değeri, bir sınıflandırıcının sahtekarlıkları gerçek işlemlerden daha yüksek sıralama olasılığı olarak yorumlanabilir [37]. Sahtekarlık tespitinde sıklıkla kullanılan bir diğer sıralama ölçütü de kesinlik-geri çağırma eğrisinin altındaki alana karşılık gelen ortalama öngörüdür [23]. Bu ölçütler tespitte yaygın olarak kullanılsa da

$$^{k} \qquad \Gamma(t) \qquad \frac{\nu}{k'} < k \text{ise } \gamma t$$

Burada $\Gamma(t)$ CP'nin maksimum değeridir$_k$ (t) ve γ_t t günündeki sahte kart sayısıdır. (6)'dan NCP_k $(t)'nin$ [0, 1] aralığında değerler aldığını, CP_k $(t)'$nin ise
$\gamma_t > k$ olduğunda [0, 1] ve aksi takdirde [0, (γ_t /k)] içinde. Örneğin,

sorunlar, maliyet temelli ölçümler özellikle dolandırıcılık tespiti amacıyla tasarlanmıştır. Maliyet tabanlı ölçümler
res [6], [47], [55] bir dolandırıcılığın parasal kaybını, karışıklık matrisinin her bir girdisiyle bir maliyet ilişkilendiren bir maliyet matrisi aracılığıyla ölçmektedir. Elkan [29] bir maliyet matrisinin yanıltıcı olabileceğini göstermektedir çünkü minimum/maksimum kayıp

problem zaman içinde değişebilir. Bu sorunu önlemek için, maksimum kayba göre performansı değerlendirmek üzere *normalleştirilmiş* maliyet [66] veya tasarruf [6] kullanılır.

Performans ölçümlerinin, FDS tarafından ortaya çıkarılan tüm uyarıları kontrol etmeleri gerektiğinden, araştırmacıların erişilebilirliğini de hesaba katması gerektiğini savunuyoruz. Araştırmacıların sahip olduğu sınırlı zaman göz önüne alındığında, her gün yalnızca birkaç uyarı doğrulanabilir ve bu nedenle etkili bir FDS, araştırmacılara az sayıda güvenilir uyarı sağlamalıdır. Bölüm III'te açıklanan uyarı hassasiyeti ölçütlerini ortaya koymamızın nedeni budur.

C. Gerçek Dünya FDS'sinde Ele Alınması Gereken Başlıca Zorluklar

Bölüm I'de öngörüldüğü üzere, bir FDS tasarlanırken ele alınması gereken başlıca zorluklar şunlardır: 1) meşru işlemler hileli işlemlerden çok daha fazla olduğu için sınıf *dengesizliğinin üstesinden gelmek*; 2) hem hileli hem de gerçek işlemlerin istatistiksel özellikleri zamanla değiştiği için *kavram kaymasının üstesinden gelmek;* ve 3) müfettişlerin geri bildirimi şeklinde sağlanan az sayıda yeni denetlenmiş işlemle çalışmak.

1) Sınıf Dengesizliği: Kredi kartı işlemlerinde sınıf dağılımı son derece dengesizdir, çünkü [45] ve [24]'te ve bizim analizimizde gösterildiği gibi dolandırıcılıklar tipik olarak toplam işlemlerin %1'inden daha azdır (bkz. Tablo I). Sınıf dengesizliği altında öğrenme son zamanlarda çok ilgi görmektedir, çünkü geleneksel öğrenme yöntemleri, tespit problemlerinde kesinlikle ilgilenilen sınıf olan azınlık sınıfında düşük performans gösteren sınıflandırıcılar ortaya çıkarmaktadır. Sınıf dengesizliği ile başa çıkmak için çeşitli teknikler önerilmiştir ve kapsamlı bir genel bakış için okuyucuyu [38]'e yönlendiriyoruz. Sınıf dengesizliği ile başa çıkmak için iki ana yaklaşım vardır:

1) *örnekleme* yöntemleri ve 2) *maliyet tabanlı yöntemler*. Örnekleme yöntemleri, geleneksel bir öğrenme algoritmasını çalıştırmadan önce eğitim setindeki sınıf dağılımını dengelemek için kullanılırken, maliyet tabanlı yöntemler azınlık sınıfına daha büyük bir yanlış sınıflandırma maliyeti atamak için öğrenme algoritmasını değiştirir [29]. Örnekleme yöntemleri, çoğunluk sınıfından örnekleri kaldırarak eğitim setindeki sınıf oranlarını dengeleyen alt örnekleme ve azınlık sınıfının eğitim örneklerini çoğaltarak aynı amaca ulaşan aşırı örnekleme olarak ikiye ayrılır [21]. SMOTE [17] gibi gelişmiş aşırı örnekleme yöntemleri, örnek çoğaltma yerine enterpolasyon yoluyla azınlık sınıfından sentetik eğitim örnekleri oluşturur. Maliyet tabanlı yöntemler, azınlık ve çoğunluk sınıfına ait örneklerdeki sınıflandırma hataları için farklı kayıpları dikkate aldıklarından, eğitim verilerinin oranını dengelemeye ihtiyaç duymazlar. Kredi kartı dolandırıcılığı tespitinde, gözden kaçan bir dolandırıcılığın maliyetinin genellikle işlem tutarıyla orantılı olduğu varsayılır [6], [47], [55] ve bu, dolandırıcılıklara daha büyük bir yanlış sınıflandırma maliyeti atar, böylece sınıflandırıcıyı bir dolandırıcılığı gözden kaçırma riskini almak yerine yanlış uyarıları tercih etmeye yönlendirir. Sonuç olarak, araştırmacılar kesin uyarılara ihtiyaç duyarken bu algoritmalar birçok yanlış pozitif üretebilir.

2) Kavram Kayması: Literatürde genellikle kavram kayması olarak adlandırılan kredi kartı işlemleri akışında değişikliklere/evrimlere yol açan iki ana faktör vardır [27], [35]. İlk olarak, gerçek işlemler gelişir çünkü

Kart sahipleri genellikle harcama davranışlarını zaman içinde değiştirirler (örneğin, tatil dönemlerinde yılın geri kalanından farklı olarak daha fazla alışveriş yaparlar). İkinci olarak, yeni dolandırıcılık faaliyetleri gerçekleştirildiği için dolandırıcılıklar zaman içinde değişir. Deneylerimizde (bkz. Bölüm VI-D), gerçek dünyadaki e-ticaret işlemlerinden oluşan iki büyük veri setinde kredi kartı işlemlerinin değişen doğasını gözlemliyoruz. Kavram kayması altında öğrenme, veri güdümlü yöntemlerin karşılaştığı en büyük zorluklardan biridir, çünkü bu koşullarda çalışan sınıflandırıcılar, pratikte en alakalı güncel denetimli bilgileri otonom olarak tanımlarken, eski olanları göz ardı etmek zorundadır. Kavram kayması uyarlama yaklaşımları iki aileye ayrılabilir: 1) aktif adaptasyon ve 2) pasif adaptasyon.

Aktif yaklaşımlar [4], [9], [34], [50], [60] sınıflandırma hatasını ve/veya veri dağılımını analiz ederek gelen verileri izlemek için bir değişiklik tespit testi [3] veya diğer istatistiksel tetikleyiciler kullanır [2]. Gelen verilerde bir değişiklik tespit edilir edilmez adaptasyon etkinleştirilir ve sınıflandırıcı, sürecin mevcut durumuyla uyumlu olduğu düşünülen son denetimli örnekler üzerinde güncellenir/yeniden eğitilir. Bu nedenle, aktif yaklaşımlar çoğunlukla veri dağılımı aniden değiştiğinde ve verileri üreten süreç bir dizi durağan durumdan geçtiğinde uygundur.

Pasif yaklaşımlarda sınıflandırıcı, yeni denetimli örnekler mevcut olduğunda, herhangi bir açık tetikleme mekanizması içermeden sürekli olarak güncellenir. Topluluk yöntemleri [23], [30], [43], [61], [72] ve son denetimli örneklerin kayan penceresi üzerinden eğitilen sınıflandırıcılar (STAGGER [57] ve FLORA [67] gibi) muhtemelen en kapsamlı şekilde araştırılan pasif çözümlerdir. Pasif yaklaşımlar, kademeli olarak sürüklenen o r t a m l a r d a ve denetimli bilgi gruplar halinde sağlandığında daha uygun olanlardır.

Veri akışları hem kavram kayması hem de dengesiz dağılımlarla karakterize edildiğinde, adaptasyon genellikle topluluk yöntemlerini ve yeniden örnekleme tekniklerini birleştirerek elde edilir [26], [36], [64]. Alternatif bir yaklaşım, azınlık sınıfının eğitim örneklerini zaman içinde yaymak [36] ve muhtemelen çoğunluk sınıfını daha az örneklemekten oluşur. Chen ve He [18], yalnızca mevcut kavrama ait azınlık sınıfından örnekleri yayan REA'yı önermiştir.

3) Uyarı-Geri Bildirim Etkileşimi ve Örnek Seçim Yanlılığı: Literatürde kredi kartı dolandırıcılığı tespiti için kullanılan sınıflandırıcıların çoğu (bkz. [11], [12], [15]), işlem etiketlerinin işlemin onaylanmasının hemen ertesi günü mevcut olduğu varsayılan deneylerde test edilmiştir. Gerçek dünyadaki bir FDS'de (Bölüm II-C), tek yeni denetimli bilgi, araştırmacılar tarafından sağlanan F_t geri bildirimleridir, ancak her gün yetkilendirilen işlemlerin büyük çoğunluğu kısa bir süre içinde etiket alamaz ($|F_t| << |T_t|$). Geri bildirimler, iki ana nedenden dolayı her gün işlenen işlemleri temsil etmemektedir: 1) geri bildirimler, dolandırıcılık olma olasılığı yüksek olan işlemleri içerir ve
2) Geri bildirimlerdeki sahtekarlık oranı, her gün meydana gelen sahtekarlık oranından farklıdır. Dolayısıyla, geri bildirimler bir tür önyargılı eğitim setini temsil eder: bu sorun literatürde SSB olarak bilinen durumu çağrıştırır [19].

Eğitim verileri test verilerinin dağılımına uymadığından, taraflı bir eğitim seti öğrenme algoritmalarının performansını engelleyebilir. Okuyucu, Örnek Seçim Yanlılığı (SSB) üzerine bir araştırma için [49]'a başvurabilir. Burada basitçe üç farklı SSB türü olduğundan bahsedeceğiz**!** (SSB'ler**!**): sınıf-öncelikli yanlılık, özellik yanlılığı (ortak değişken kayması olarak da adlandırılır) ve tam yanlılık. SSB için standart bir çözüm *önem ağırlıklandırmasıdır* [32], [69], [70], yani test setindeki veri dağılımına daha çok benzeyen eğitim örneklerine daha büyük ağırlıklar atayan yarı denetimli yeniden ağırlıklandırma tekniklerdir. Önem ağırlıklandırmanın ana fikri, öğrenme sürecinde en önyargılı örneklerin etkisini azaltmaktır. SSB'yi düzeltmek için sınıflandırıcı toplulukları da önerilmiştir [31].

FDS (uyarı veren) ve araştırmacılar (gerçek etiketleri sağlayan) arasındaki etkileşim, az sayıda-çok bilgilendirici-örnek seçmenin ve bunların etiketlerini FDS'de araştırmacılar olacak bir kahine sormanın mümkün olduğu aktif öğrenme senaryosunu [58] hatırlatmaktadır. Ancak bu gerçek dünyadaki bir FDS'de mümkün değildir çünkü araştırmacılar en fazla sayıda sahtekarlığı tespit etmek için en şüpheli işlemlere odaklanmak zorundadır. Bilgilendirici örnekler elde etmek için (muhtemelen gerçek) işlemlerin kontrol edilmesine yönelik talepler göz ardı edilecektir. Araştırmacıların kontrol edebileceği sınırlı sayıda işlem göz önüne alındığında, bu soruların ele alınması, bazı yüksek riskli işlemlerin kontrol edilmediği ve bunun sonucunda tespit performansında kayıp olduğu anlamına gelecektir.

V. Önerilen Öğrenme Stratejisi

Geri bildirimlerin (F_t) ve gecikmeli örneklerin ($D_{t-\delta}$) çok farklı denetimli örnek kümeleri olduğunu vurgulamak önemlidir. İlk fark oldukça belirgindir: F_t son güncel bilgileri sağlarken, $D_{t-\delta}$ ertesi gün onaylanacak işlemleri analiz etmeyi amaçlayan bir sınıflandırıcıyı eğitmek için zaten eski olabilir. İkinci fark

F_t ve $D_{t-\delta}$ 'deki sahtekarlıkların yüzdesi ile ilgilidir: $D_{t-\delta}$ 'deki sınıf oranı büyük ölçüde gerçek sınıfa doğru çarpıkken (Tablo I'deki sahtekarlık oranlarına bakınız)

F_t 'daki sahtekarlık sayısı aslında K_{t-1} 'un tespit performansına bağlıdır ve yüksek hassasiyet değerleri F_t 'un sahtekarlıklara doğru çarpık olmasına bile neden olabilir. Üçüncü ve muhtemelen en ince fark, F_t 'daki denetimli çiftlerin bağımsız olarak çekilmemesi, bunun yerine K_{t-1} tarafından dolandırılmış olma olasılığı en yüksek olanlar olarak seçilen kartlardan yapılan işlemler olmasıdır. Bu nedenle, F_t SSB'den etkilenir ve F_t üzerinde eğitilen herhangi bir sınıflandırıcı prensipte dolandırıcılık ihtimali en yüksek olan işlemleri nasıl etiketleyeceğini öğrenir. Dolayısıyla, gerçek işlemlerin büyük çoğunluğunda bu prensipte kesin olmayabilir. Sezgilerimiz, geri bildirimlerin ve gecikmeli örneklerin

iki farklı sınıflandırma problemini temsil eder ve

bu nedenle ayrı ayrı ele alınmaları gerekir. Bu nedenle, öğrenme stratejimiz, yalnızca geri bildirimler (yani F_t) ve yalnızca gecikmeli denetimli örnekler (yani D_t) üzerinde bir sınıflandırıcı eğitmekten ve hangi işlemlerin uyarılacağını belirlemek için $_{PKt}(+|x_i)$ tanımlanırken ve sınıflandırıcı, geri bildirimler veya gecikmeli örnekler gibi mevcut en son denetimli çiftleri içeren bir yığın üzerinde her gün güncellenir. Bölüm III'te olduğu gibi, δ günlük sabit bir doğrulama gecikmesini dikkate alıyoruz. Özellikle, işlemek için

$t + 1$. günde yetkilendirilen işlemler, Q günlük geri bildirimlere $\{F_t, \ldots, F_{t-(Q-1)}\}$, ve M günlük gecikmeli denetimli örnekler $\{D_{t-\delta}, \ldots, D_{t-(\delta+M-1)}\}$, ve bu sonuncular açıkça aynı günlerde alınan geri bildirimleri içerir (yani, $F_i \subset D_i, i \leq t - \delta$). Algoritma 1'de ayrıntılı olarak açıklanan öğrenme stratejimiz, ayrı ayrı a

sınıflandırıcı F_t geri bildirimler üzerine

$$F_t = \text{TRAIN}(\{F_t, \ldots, F_{t-(Q-1)}\}) \quad (7)$$

ve gecikmeli denetimli örnekler üzerinde bir sınıflandırıcı

$$D_t = \text{TRAIN}(\{D_{t-\delta}, \ldots, D_{t-(\delta+M-1)}\}) \quad (8)$$

ve sonsal olasılığı aşağıdaki gibi tanımlanan A_t *toplulaştırma* sınıflandırıcısı ile sahtekarlıkları tespit etmek için

$$_{PAt}(+|x) = {}_{\alpha PFt}(+|x) + (1 - \alpha)_{PDt}(+|x) \quad (9)$$

Burada $0 \leq \alpha \leq 1$, F_t ve D_t katkılarını dengeleyen ağırlık parametresidir. Böylece, $t + 1$ gününde yetkilendirilen işlemleri uyaran K_t sınıflandırıcısının sonsal olasılığı (9) ile verilir.

Q ve M parametreleri, sırasıyla

Sınıflandırıcılarımızı eğitmek için kaç günlük geri bildirim ve gecikmeli denetimli örnek kullanılacağı, toplam geri bildirim sayısı ve sahtekarlık yüzdesi dikkate alınarak belirlenmelidir. F_t eğitim seti yaklaşık olarak $Q - |F_t|$ örnekleri içerir (her gün farklı sayıda geri bildirim sağlanabilir) ve bu, bir sınıflandırıcıyı eğitmek için yeterince büyük bir sayı olmalıdır.

sınıflandırıcı, yüksek boyutlarda oldukça zorlu bir sınıflandırma problemini ele alır. Ancak, Q eski geri bildirimleri içermeyecek şekilde keyfi olarak büyük yapılamaz. Benzer hususlar, yeterli sayıda sahtekarlık içermesi gereken gecikmeli işlemleri içeren kabul edilen gün sayısı olan *M'yi* ayarlarken de geçerlidir. Yine de F_t eğitim kümesine δ günden önce alınan geri bildirimleri dahil etmenin mümkün olduğunu unutmayın ($Q \geq \delta$)

ve özellikle deneylerimizde $Q = \delta + M$ kullandık.

Önerilen öğrenme stratejisinin arkasındaki mantık şudur

iki yönlüdür. İlk olarak, bir sınıflandırıcıyı (7) yalnızca geri bildirimler üzerinde eğiterek, aksi takdirde gecikmeli denetimli örneklerden sayıca fazla olacak olan bu denetimli örneklere daha fazla alaka gösterilmesini garanti ediyoruz. İkinci olarak, yalnızca hem F_t hem de D_t *'nin* büyük olasılıkla dolandırıcılık olarak değerlendirdiği işlemleri uyarıyoruz: bu, pratikte, her gün işlenen çok sayıda işlem nedeniyle, uyarıların $_{PAt'}$n bire çok yakın değerlerine karşılık geldiği gerçeğinden kaynaklanmaktadır. İzin verin

SSB'den etkilendiğini hatırlayalım

önceki olasılıklarını toplamaktan oluşur.

Aşağıda, adaptasyonun pasif bir yaklaşıma göre gerçekleştirildiği önerilen öğrenme stratejisini detaylandırıyoruz

uyarı-geri bildirim etkileşimi. SSB'den etkilenmeyen tek eğitim örneği gecikmeli denetimli örneklerdir, ancak bunlar kavram kayması nedeniyle eski olabilir.

A. Önerilen Öğrenme Stratejisinin Uygulanması

Deneylerimizde, önerilen öğrenme stratejisini D_t öğrenmeye yönelik iki ana akım yaklaşıma karşılık gelen iki farklı senaryoda uyguluyoruz. İlkinde, D_t

Algoritma 1 Önerilen Öğrenme Stratejisi

Gereklilik: M ve Q, yani sırasıyla kullanılacak gecikmiş örneklerin ve geri bildirimlerin gün sayısı; F_t ve D_t daha önce eğitilmiş sınıflandırıcılar.

$T_{t+1} \leftarrow t + 1$ günündeki işlemler.

her $x \in T$ işlemi **için**$_{t+1}$ **do**

$_{PFt}(+, x)$ hesaplayın

$_{PDt'yi}$ hesaplayın $(+, x)$

$_{PAt}(+, x)'i$ (9)'daki gibi hesaplayın

T_{t+1} $'yi\ P_{At}(+, -)'ye$ göre sıralayın, A_t uyarılarını oluşturun.

sınıflandırıcıyı güncellerse

$F_{t+1} \leftarrow A'$da uyarılan kartlardan gelen geri bildirimler$_t$.

$F_{t+1} \leftarrow \text{TRAIN}(\{F_{t+1}, \ldots, F_{t-Q}\})$

$D_{t+1-\delta} \leftarrow t + 1$'de yetkilendirilen işlemler $- \delta$

$_{Dt+1} \leftarrow \text{TRAIN}(_{\{Dt+1-\delta, \ldots, Dt-(\delta+M)\}})$

(9)'da tanımlandığı gibi F_t, D_t ve A_t **döndürür.**

W^D ile gösterdiğimiz [62] ve [63]'teki gibi bir *kayan pencere sınıflandırıcısı*$_t$ iken, ikincisinde D_t, E^D ile gösterdiğimiz [36] ve [23]'e benzer bir sınıflandırıcı *topluluğudur*.$_t$ Hem W^D hem de E^D sınıflandırıcıları gecikmeli örnekler üzerinde eğitilir.

$_t$ $_t$

ples $\{D_{t-\delta}, \ldots, D_{t-(\delta+M-1)}\}$. Ancak, W^D bu$_t$amaç için tek bir model kullanırken, E^D M_i sınıflandırıcıdan $\{M_1, M_{12}, \ldots, M_M\}$ oluşan bir topluluktur; burada her bir sınıflandırıcı M_i farklı bir günün gecikmeli örnekleri, yani $D_{t-\delta-i}$, $i = 0, \ldots, M-1$ üzerinde eğitilir. Sonsal $_{PED}(+|x)$, bireysel sınıflandırıcıların sonsal^t olasılıklarının ortalaması alınarak elde edilir, yani $P_D(+|x) = (\ ^M P_{Mi}(+|x))/(M)$.

Kayan pencere durumunda, önerilen öğrenme stratejisi egy, sınıflandırıcının son durumunun analiz edilmesinden oluşur F_t ve W^D 'yi bir araya getiren A^W, yani $P_W(+|x) =$

$_{PFt}(+|x) + (1 - \alpha)_{PWD}(+|x)$ (9)'daki gibi. Karşılaştırma ölçütü A ile karşılaştırmak için$_W$ $_t$ W_t, eğitilmiş olan $_t$ sınıflandırıcıdır

Aynı zaman aralığına atıfta bulunan tüm denetimli işlemlerde (böylece gecikmeli örnekler ve geri bildirimler karıştırılır): $\{Ft, \ldots, Ft-(\delta-1), Dt-\delta, \ldots, Dt-(\delta+M-1)\}$.

Benzer şekilde, topluluk durumunda, önerilen öğrenme stratejisi, sınıflandırıcının son durumunun analiz edilmesinden oluşur A^E, F'nin son değerlerinin toplanmasıyla elde edilir. ve E^D, yani $_{PAE}(+|x) = _{\alpha PFt}(+|x) + (1 - \alpha)P_{ED}(+|x)$,

$_t$ $_t$

(9)'daki gibi. A ile karşılaştırılacak ölçüt^E $_t$ bu

classifier E_t, whose individuals are $\{M_1, M_2, \ldots, M_M, F_t\}$, and whose posterior $P_{E_t}(+|x)$ is estimated by averaging the posterior probabilities of all its individuals, i.e., $P_{E_t}(+|x) = (\ _i^M P_{M_i}(+|x) + P_F(+|x))/(M + 1)$.

$_{At}$ ve $_{At}$ toplamalarının^W her ikisinde^E de $\alpha = 0{,}5$ olarak belirlenmiştir.

olarak geri beslemeye ve gecikmeli sınıflandırıcıya eşit katkı sağlar.

Bölüm VI-F'de daha iyi tartışılmıştır. İlgili tüm temel

$_t$

TABLO I
VERİ
SETLERİ

Id	Start day	End day	# Days	# Instances	# Features	% Fraud Trx
2013	2013-09-05	2014-01-18	136	21'830'330	51	0.19%
2014-2015	2014-08-05	2015-05-31	296	54'764'384	51	0.24%

aynı dolandırıcılık örnekleri. Bu az örnekleme stratejisi, dengeli dağılıma sahip ağaçların öğrenilmesine ve çoğunluk sınıfının birçok alt kümesinden yararlanılmasına olanak tanır. Aynı zamanda, bu sınıflandırıcıların eğitim süreleri oldukça düşüktür. Alt örneklemenin bir dezavantajı, veri setinden ilgili eğitim örneklerini potansiyel olarak çıkarmamızdır, ancak bu sorun, her bir temel sınıflandırıcı için 100 farklı ağaç öğrenmemiz gerçeğiyle hafifletilmektedir.

VI. DENEYLER

Deneylerimiz aşağıdaki şekilde düzenlenmiştir. Bölüm VI-A'da veri setlerini tanımlıyoruz ve Bölüm VI-B'de deneysel ayarları detaylandırıyoruz. Bölüm VI-C, önerilen öğrenme stratejisinin etkinliğini değerlendirmek için Bölüm V-A'da açıklanan sınıflandırıcıları kullanan ilk deneyimizi sunmaktadır. İkinci deneyde (Bölüm VI-D), daha fazla analiz yapıyoruz

54 milyondan fazla kredi kartı işleminden elde edilen

10 aydır ve bu akışın kavram kaymasından ciddi şekilde etkilendiğini göstermektedir. Ardından, önerilen öğrenme stratejisinin adaptasyon yeteneğini araştırmak için, işlem akışının belirli konumlarına sentetik olarak ani bir kavram kayması ekliyoruz ve sınıflandırma performansını değerlendiriyoruz. Üçüncü deneyde (Bölüm VI-E), uyarı-geribildirim etkileşiminin getirdiği örnek seçim yanlılığını gelenekselbir teknik gösteriyoruz ağırlıklandırmanın [19]

doğru SSB- geri bildirimlerin eğitim setleri üzerinde etkili değildir.

Son olarak, Bölüm VI-F'de, önerilen öğrenme stratejisini etkileyen en önemli parametreleri tartışıyoruz.

A. Veri Setlerimiz

Endüstriyel ortağımız tarafından sağlanan Avrupalı kredi kartı sahiplerinin çevrimiçi e-ticaret işlemlerinden oluşan iki büyük veri setini kullanıyoruz. Bu işlemler fiziksel bir terminalden başlatılmamış olsa da Şekil 1'de açıklanan aynı süreçten geçmektedir. Tablo I'de, bu işlemlerle ilgili tüm bilgileri sunuyoruz.

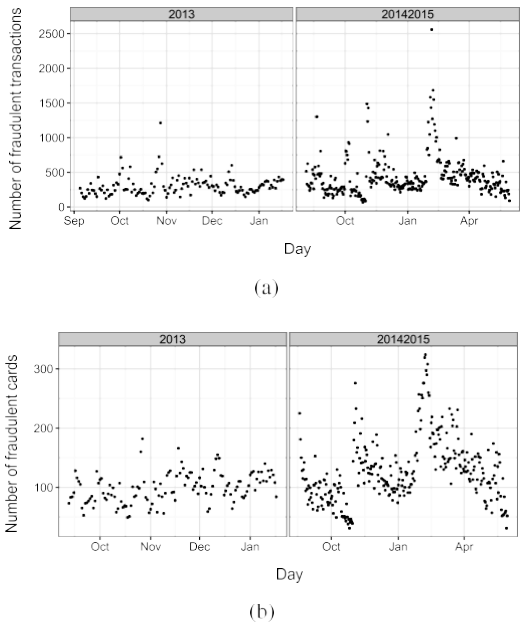
2013 ve 2014-2015 olarak gösterdiğimiz veri setleri ve

özellikle, sahtekarlıklar nedeniyle aşırı sınıf dengesizliğini vurguluyoruz

sınıflandırıcılar için (yani, F_t, W^D, W_t, M_i, $i = 1, \ldots, M$), her biri 100 ağaca sahip RF [13] benimsenmiştir. Her ağaç, ilgili eğitim setindeki tüm azınlık sınıfı örneklerini korurken çoğunluk sınıfını rastgele küçülterek elde edilen dengeli bir bootstrap örneği üzerinde eğitilir. Bu şekilde, her ağaç rastgele seçilen gerçek işlemler üzerinde eğitilir ve

tüm işlemlerin yaklaşık %0,2'sini oluşturmaktadır. Şekil 3'te gösterildiği gibi, günlük dolandırıcılık sayısı zaman içinde önemli ölçüde değişmektedir ve bazen aynı kart üzerinde birden fazla dolandırıcılık yapıldığını gösteren, dolandırıcılık yapılan kartlardan daha fazla dolandırıcılık işlemi vardır. 2013 veri seti [20]'de de kullanılmış ve bu veri setinin bir kısmı uygun şekilde anonimleştirilerek indirilmek üzere kamuya açık hale getirilmiştir [22].

Sahtekarlık tespit performansını P açısından güvenilir bir şekilde değerlendirmek için$_k$, *CARD_ID* bileşenini tüm özellik vektörlerinden çıkardık. Bu, bir sınıflandırıcıyı geçmiş işlemlerden oluşan bir veri kümesi üzerinde test ederken çok önemlidir, çünkü *CARD_ID* değişkenini girdi olarak alan bir sınıflandırıcı, bunu birden fazla sahtekarlığı tespit etmek için ayırt edici bir özellik olarak öğrenebilir.



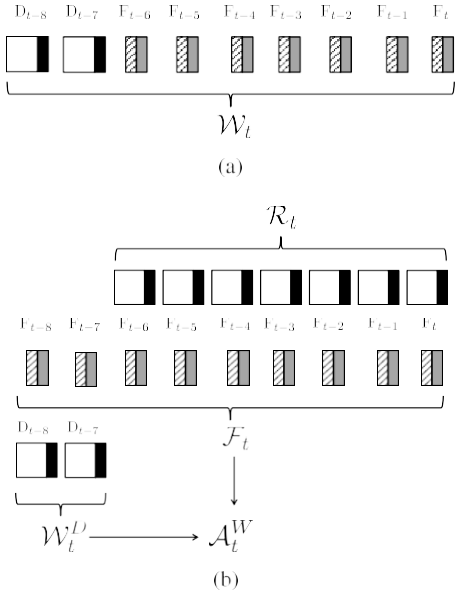
Şekil 3. Tablo I'de açıklanan veri setlerinde günlük hileli işlem ve kart sayısı. Kartlardan daha fazla hileli işlem olduğu ortaya çıkmaktadır, bu da bazı kartların bir hileden daha fazlasını aldığı anlamına gelmektedir.
(a) Sahte işlem sayısı. (b) Sahte kart sayısı.

aynı kartın farklı günlerde kullanılması (dolayısıyla çok iyimser bir performans sağlanması). Bununla birlikte, gerçek dünyadaki bir FDS'de, Bölüm II'de tartışıldığı gibi, kart hemen bloke edildiğinden, ilkini tespit ettikten sonra aynı karttan birden fazla dolandırıcılık olması mümkün değildir. Farklı bir seçenek, ilk sahtekarlığı tespit ettikten sonra aynı kartın tüm işlemlerini kaldırmak olabilir. Ancak bu, mevcut dolandırıcılık sayısını azaltacak ve veri setimizdeki sınıf dengesizliğini daha da kötüleştirecektir. Bu nedenle, toplu özellikleri hesaplamak için yalnızca *CARD_ID'yi* dikkate alıyoruz ve özellik vektörlerine dahil etmiyoruz.

B. Deneysel Ayarlar

Endüstriyel ortağımızla anlaşarak, müfettişlerin her gün DDM tarafından uyarılan 100 kartı kontrol edebileceğini varsaydık. Böylece, F_t her gün 100 farklı kart sahibinden gelen her bir uyarılmış işlemi içeren Q gün boyunca eğitilir. Geri bildirimlerin etiketleri talep eden gerçek sınıflandırıcıya bağlı olduğunu hatırlayalım. Bu nedenle, F_t eğitim seti A_t içinde kullanıldığında ve tek başına kullanıldığında farklı olabilir: aslında, ilk durumda, uyarılar D_t 'nin posterioruna da bağlıyken, ikincisinde uyarılar benzersiz bir şekilde F_t tarafından belirlenir.

Veri setlerimizdeki genel dolandırıcılık tespit performansını hem günlük performans ölçümlerinin (P_k , CP_k ve AUC) ortalamasını alarak hem de sınıflandırıcıların her bir gündeki sıralamalarının toplamını analiz ederek değerlendiriyoruz. Özellikle, her j gününde test edilen S sınıflandırıcıyı en iyi performans gösterenden en düşük performans gösterene doğru sıralıyoruz ve $r_{K,j} \in \{1, \ldots, S\}$ K sınıflandırıcısının j günündeki sıralaması: K en iyi sınıflandırıcı olduğunda, sıralaması maksimum olduğunda, yani $r_{K,j} = S$, en kötü olduğunda ise $r_{K,j} = 1$'dir. 25]'te önerildiği gibi, bir



Friedman testi [33] gerçekleştiriyoruz ve tüm değerlerin aynı olduğu boş hipotezini reddediyoruz.

Şekil 4. Deneylerimizde dikkate alınan sınıflandırıcılar tarafından kullanılan denetimli bilgi. Bu açıklayıcı örnekte $\delta = 7$, $M = 2$ ve $Q = 7 + 2 = 9$. (a) Tüm etiketli işlemlerin bir araya getirilmesi. (b) Geri bildirimleri ve gecikmeli örnekleri ayırma.

sınıflandırıcılar aynı performansı elde eder. Ardından, tüm günlük sıralamaları toplayarak global bir sıralama tanımlarız (bkz. Tablo II): sıralamaların toplamı ne kadar büyükse sınıflandırıcı o kadar iyidir ve global sıralamadaki farklılıkların anlamlı olup olmadığını belirlemek için eşleştirilmiş t-testleri kullanırız. Uygulamada, her bir K ve H sınıflandırıcı çifti için, sıralamalarını karşılaştırmak için bir t-testi kullanılır

tüm günler üzerinden (yani, $r_{K,j} - r_{H,j}, j \in \{1, \ldots, J\}$), J gün sayısı olmak üzere.[3]

Performansın değişkenliğini azaltmak için her deney 10 kez tekrarlanmıştır ve sınıflandırıcıları birden fazla günde karşılaştırırken sınıflandırıcı notundan t indeksini çıkarıyoruz. Deneylerimizin çoğunda, bir haftalık doğrulama gecikmesini ($\delta = 7$) ve $M = 8$'i dikkate alıyoruz, öyle ki

Kullanılan toplam geri bildirim sayısı $Q = M + \delta = 15$'tir.

Bölüm VI-F'de, daha uzun bir doğrulama gecikmesi $\delta = 15$ ve $M = 15$, $Q = 30$ dikkate alınarak deneyleri tekrarlıyoruz.

C. Gecikmeli Denetimli Örneklerden Geri Bildirimleri Ayırma

Önerilen öğrenme stratejisinin etkinliğini değerlendirmek için, önerilen sınıflandırıcıların A^W (resp. A^E) performansını Bölüm V-A'da tanıtılan ilgili ölçütlerle ve bunların posteri- orlarını, yani F ve W^D (resp. E^D) tanımlamak için kullanılan sınıflandırıcılarla karşılaştırıyoruz. Şekil 4, A^W ve ilgili sınıflandırıcılar kullanılırken kullanılan eğitim setini gösterirken, Tablo III en önemli parametreleri ve dikkate alınan sınıflandırıcılar tarafından kullanılan eğitim örneklerini özetlemektedir.

Bu deneye, t ve t-δ günleri arasında yetkilendirilen tüm işlemler üzerinde eğitilen *ideal* sınıflandırıcı R_t da dahil edilmiştir. Bu sınıflandırıcı ideal bir muadil olarak kabul edilir

t

3Ardışık günlerde kullanılan eğitim setleri büyük ölçüde örtüştüğü için sıralamalar bağımsız değildir ve bu durum testlerin anlamlılığını etkileyebilir. Aslında, bir sınıflandırıcı bir günde diğerlerinden daha iyi performans gösterdiğinde, aynı şeyin sonraki birkaç gün boyunca da gerçekleşmesi muhtemeldir. Ancak bu, Friedman testi gibi parametrik olmayan testler için standart bir *post hoc* analizidir [25].

TABLO II

DOLANDIRICILIK TESPIT PERFORMANSI 15 GÜNLÜK İŞLEMLER KULLANILDIĞINDA (Δ = 7, M = 8 VE Q = 15)

Classifier	Dataset	Average P_k			Average CP_k			Average AUC		
		mean (std)	sum of ranks	comparison	mean (std)	sum of ranks	comparison	mean (std)	sum of ranks	comparison
$\mathcal{A}^W$	2014-2015	0.77 (0.21)	1796.50	a	0.37 (0.18)	1824.00	a	0.94 (0.02)	1396.00	b
$\mathcal{F}$	2014-2015	0.73 (0.23)	1632.00	b	0.32 (0.17)	1505.00	b	0.87 (0.05)	409.00	e
$\mathcal{R}$	2014-2015	0.63 (0.24)	1156.00	c	0.30 (0.18)	1354.50	c	0.96 (0.02)	1822.00	a
$\mathcal{W}$	2014-2015	0.61 (0.25)	1055.50	d	0.25 (0.14)	955.00	d	0.91 (0.04)	865.00	d
$\mathcal{W}^D$	2014-2015	0.57 (0.26)	889.00	e	0.25 (0.14)	885.00	e	0.94 (0.03)	1315.00	c
$\mathcal{A}^W$	2013	0.75 (0.20)	732.00	a	0.35 (0.12)	754.50	a	0.94 (0.03)	631.00	b
$\mathcal{F}$	2013	0.73 (0.21)	693.00	b	0.32 (0.13)	670.50	b	0.89 (0.05)	229.00	e
$\mathcal{R}$	2013	0.58 (0.22)	493.50	c	0.25 (0.11)	514.00	c	0.96 (0.01)	736.00	a
$\mathcal{W}$	2013	0.54 (0.25)	434.00	d	0.22 (0.11)	387.00	d	0.91 (0.05)	355.00	d
$\mathcal{W}^D$	2013	0.50 (0.23)	345.00	e	0.21 (0.09)	330.00	e	0.93 (0.03)	539.00	c
$\mathcal{A}^E$	2014-2015	0.77 (0.21)	981.50	a	0.39 (0.17)	940.00	a	0.94 (0.03)	873.00	b
$\mathcal{F}$	2014-2015	0.73 (0.23)	827.50	b	0.36 (0.17)	800.50	b	0.87 (0.06)	294.00	d
$\mathcal{E}$	2014-2015	0.66 (0.25)	637.50	c	0.26 (0.14)	533.50	c	0.94 (0.03)	943.00	a
$\mathcal{E}^D$	2014-2015	0.54 (0.26)	323.50	d	0.23 (0.12)	276.00	d	0.93 (0.03)	660.00	c
$\mathcal{A}^E$	2013	0.76 (0.20)	410.50	a	0.37 (0.14)	335.00	a	0.94 (0.02)	380.00	a
$\mathcal{F}$	2013	0.73 (0.21)	354.00	b	0.35 (0.15)	285.00	b	0.89 (0.04)	129.00	c
$\mathcal{E}$	2013	0.62 (0.23)	246.50	c	0.24 (0.11)	193.00	c	0.93 (0.03)	374.00	a
$\mathcal{E}^D$	2013	0.48 (0.24)	119.00	d	0.20 (0.11)	97.00	d	0.93 (0.03)	247.00	b

TABLO III

DENEYLERIMIZDE DIKKATE ALINAN SINIFLANDIRICILAR

Symbol	supervised samples	adaptation	# days training
$\mathcal{F}$	feedbacks	sliding	Q
$\mathcal{W}^D$	delayed	sliding	M
$\mathcal{W}$	feedbacks + delayed	sliding	$\delta + M$
$\mathcal{A}^W$	feedbacks + delayed	sliding	$Q + M$
$\mathcal{R}$	all the recent	sliding	δ
$\mathcal{E}^D$	delayed	ensemble	M
$\mathcal{E}$	feedbacks + delayed	ensemble	$\delta + M$
$\mathcal{A}^E$	feedbacks + delayed	ensemble	$Q + M$

gerçekçi olmayan bir şekilde müfettişlerin her yetkili işleme her gün doğru etiketi atayabileceğini varsayan kayan pencere sınıflandırıcıları. Özellikle, R_t eğitim seti uyarı-geri bildirim etkileşiminden etkilenmez.

Tablo II, iki veri seti için ayrı ayrı tüm gruplar üzerindeki ortalama P_k , CP_k ve AUC değerlerini göstermektedir. *Karşılaştırma* sütunları, yukarıda açıklanan sıralamalara ilişkin eşleştirilmiş t-testinin sonuçlarını bildirmektedir. Aynı harfe sahip sınıflandırıcılar önemli ölçüde farklı kabul edilemez. Her iki veri setinde de A^W , P_k ve CP_k açısından *W'den daha* iyi performans göstermektedir ve bu da geri bildirimleri ve gecikmeli örnekleri ayırmanın gerçekten de iyi bir öğrenme stratejisi olduğunu göstermektedir. Aynı sonuç dikkate alınan topluluklar için de geçerlidir, yani A^E ve E. Hem A^E hem de E bireylerinin sonsal değerlerinin ortalamasını aldığından, aralarındaki fark
sadece toplama ağırlıklarında: $A'da^E$, toplam ağırlığın %50'si $_{PF(+|x)'}$e atanır ve kalan %50 diğer bireylere eşit olarak dağıtılır. Buna karşılık, E'de tüm bireyler eşit katkıda bulunur. Aynı ilişki, kayan pencere şeklinde güncellenen A^W ve W arasında geçerli değildir. Ancak, bu durumda da şunları yapabiliriz
geri bildirimlerin çok bilgilendirici olduğu ve uyarı hassasiyetini artırmak için dikkatle değerlendirilmesi gerektiği sonucuna varılmıştır. Bu durum, $F'nin$ hem W^D hem de *W'den daha* iyi performans göstermesiyle de doğrulanmaktadır. Genel bir yorum olarak, CP_k 'nin genellikle P_k 'den daha düşük olduğunu, çünkü genellikle aynı kart üzerinde birden fazla dolandırıcılık yapıldığını belirtmek isteriz.

Tablo II ayrıca sonuçları, sınıflandırıcının son değerlerini aşağıdakiler üzerinden değerlendiren küresel bir sıralama ölçütü olan AUC açısından da rapor etmektedir

(CP_k ve P_k 'den farklı olarak) sadece ilk k'da değil tüm örneklerde. AUC açısından, ideal sınıflandırıcı R, A'dan önemli ölçüde daha iyidir^W ve F açık ara daha kötüdür, bu da F'nin tüm işlemleri sıralarken etkili olmadığını gösterir.

Bu sonuçları şu şekilde yorumluyoruz: amaç en şüpheli kartların doğru bir sıralamasını elde etmek olduğunda (örn, CP_k 'yi maksimize etmek), tahmin etmek istediklerimiz kadar riskli olan işlemlere daha büyük ağırlıklar atamalıyız, dolayısıyla A^W 'yi kullanmalıyız. Aksine, tüm günlük işlemler (çoğunlukla gerçek olan) üzerinde eğitilmiş bir sınıflandırıcı, R'nin ROC eğrisi altındaki alandan (AUC) ortaya çıktığı gibi, tüm işlemleri sıralamada daha iyidir. Tablo II'de, R'nin P_k , CP_k ve AUC açısından W^D 'den daha iyi performans gösterdiğini de görebiliriz. Bu sonuç, kredi kartı işlemleri akışının durağan olmadığını göstermektedir. Aslında, hem R'nin hem de W'nin eğitim setleri^D

sırasıyla $\delta = 7$ ve $M = 8$ ardışık gün içinde yetkilendirilen tüm işlemleri içerir. Aralarındaki en önemli fark, R'nin en son işlemler üzerinde eğitilmesi, W'deki işlemlerin ise^D δ günlük bir gecikmeyle gelmesidir. R'nin daha iyi performans göstermesi

W^D , en son işlemlerin sonraki günlerdeki dolandırıcılıkları tespit etmek için daha bilgilendirici olduğunu ve dolayısıyla işlem dağılımının durağan olmadığını gösterir.

Tablo II'de raporlanan P_k ve CP_k standart sapmaları, özellikle AUC'ninkilerle karşılaştırıldığında oldukça yüksektir. Bölüm III'te tartıştığımız ve Şekil 5'te gösterildiği gibi, CP_k (ve P_k) değerleri her gün meydana gelen dolandırıcılık sayısından büyük ölçüde etkilenmektedir. Bu sayı zaman içinde büyük dalgalanmalar gösterdiğinden (bkz. Şekil 3), böyle büyük bir dağılım beklemek mantıklıdır. Tablo II'deki sınıflandırıcılar arasındaki karşılaştırmanın, bu kadar büyük bir standart sapmaya rağmen performans açısından farklılıkların her zaman önemli olduğunu gösterdiğini belirtmek isteriz. NCP değerlerinin$_k$ (bkz. Tablo IV) bu tür dalgalanmalardan daha az etkilendiğini unutmayın.

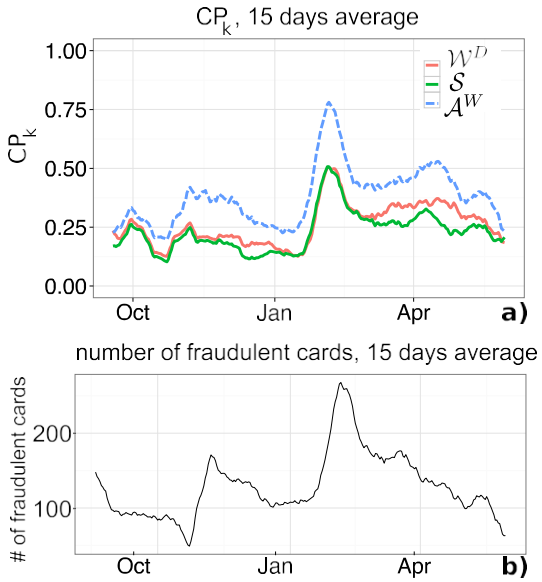
D. Konsept Sürüklenme

Bu bölümde, ilk olarak 10 ay boyunca yetkilendirilmiş 54 milyondan fazla işlemi içeren 2014-2015 veri setini analiz ediyor ve bu akışın aşağıdakilerden etkilendiğini gösteriyoruz

TABLO IV

ORTALAMA NCP_k 2013 VERI SETINDE $k \geq 100$ OLDUĞUNDA ($\delta = 15$)

classifier	mean	sd	sum of ranks	comparison	k
$\mathcal{A}^W$	0.48	0.09	506.00	a	300
$\mathcal{F}$	0.46	0.10	448.00	b	300
$\mathcal{W}$	0.38	0.11	283.00	c	300
$\mathcal{W}^D$	0.35	0.10	172.50	d	300
$\mathcal{A}^W$	0.41	0.10	519.50	a	150
$\mathcal{F}$	0.38	0.10	441.50	b	150
$\mathcal{W}$	0.29	0.10	272.50	c	150
$\mathcal{W}^D$	0.27	0.09	179.50	d	150
$\mathcal{A}^W$	0.40	0.13	518.50	a	100
$\mathcal{F}$	0.37	0.13	443.00	b	100
$\mathcal{R}$	0.29	0.10	342.50	c	100
$\mathcal{W}^D$	0.26	0.11	249.00	d	100



Şekil 5. (a) 2014-2015 veri setinde S, ^{WD} ve ^{AW} için $_{CPk}$ değerleri. (b) Aynı dönemdeki sahte kart sayısı. Görselleştirme amacıyla, bu değerlerin 15 günlük bir kayan pencere üzerinden ortalaması alınmıştır. (a)'daki $_{CPk}$ tepe noktası, (b)'deki sahte kart sayısındaki tepe noktasına karşılık gelmektedir. Bu sonuç, sınıflandırıcıların o günlerde daha hassas hale geldiğini doğrulamaktadır
çok sayıda sahte kart ile karakterize edilir.

kavram kayması. Bu amaçla, başlangıçta M günde eğitilen ve hiç güncellenmeyen (başlangıçta W^D ile çakışacak şekilde) statik bir sınıflandırıcı S_t kullanıyoruz ve bunu$_t$ (bunun ^{WD} yerine düzenli olarak güncellenir) ve $_{At}$ (güncellenmiş denetimli örneklerden de yararlanır). Durağan bir sınıflandırma probleminde, iki sınıflandırıcı S ve W^D benzer şekilde performans gösterecektir. S_t 'nin zaman içinde W^D 'den daha iyi performans göstermesi [bkz. Şekil 5(a)] bu veri setinin kavramlardan etkilendiğini doğrulamaktadır

sürüklendik. Kredi kartı işlemleri akışının durağan olmaması kulağa şaşırtıcı gelmese de, bizimki en iyi ihtimalle Bildiğimiz kadarıyla, bu kadar büyük bir işlem veri seti üzerinde kavram kaymasının etkisi üzerine yapılan ilk analizdir.

Şekil 5(a) ayrıca önerilen A^W 'nin CP_k açısından her zaman daha üstün performans gösterdiğini ve kavram kaymasına daha iyi uyum sağladığını göstermektedir. Şekil 5(a)'daki tüm veri setimizdeki hileli kartlar), tüm sınıflandırıcılar CP'nin düşük değerlerine ulaşır$_k$. Dolayısıyla Şekil 5, uyarı hassasiyetinin büyük ölçüde bir gündeki sahte kart sayısına bağlı olduğunu doğrulamaktadır. A^W 'un durağan olmayan ortamlardaki adaptasyon performansını daha fazla araştırmak için, adaptasyon yeteneklerini yapay olarak tanıtılan bir kavram kaymasına göre değerlendiriyoruz. Özellikle, daha önce tartıştığımız işlem akışını etkileyen (kademeli) sapmanın üzerine ani bir sapma ekleyerek bilinen konumlarda yapay olarak değişiklikler yapıyoruz. 20]'de olduğu gibi, ardışık olmayan iki ayda onaylanan işlemleri yan yana getirerek 10 kısa akış hazırladık. Bu akışların her birinin ortasında, yan yana gelen aylar arasındaki zaman mesafesi arttığında daha net algılanması gereken ani bir kavram kayması bulunmaktadır. Önerilen öğrenme stratejisinin adaptasyon kabiliyetini değerlendirmek için, A^W ve W^D performansını CP_k açısından karşılaştırıyoruz. Özellikle, kavram kaymasından kaynaklanan göreceli performans kaybını, ilk ve ikinci aydaki CP_k arasındaki farkın ilk aydaki CP_k değerine bölünmesiyle ölçüyoruz. Deneylerimiz, bu 10 veri setinde A^W 'nin CP_k 'sinin %7,7, W^D 'nin CP_k 'sinin ise %12,5 oranında azaldığını göstermekte ve A^W 'nin üstün adaptasyon performansını doğrulamaktadır. önerilen öğrenme stratejisi.

E. Uyarı-Geribildirim Etkileşiminden Kaynaklanan Örnek Seçim Yanlılığı

Burada, SSB'yi düzeltmek için yaygın bir çözüm olan önem ağırlıklandırmanın [19], uyarı-geribildirim etkileşiminden kaynaklanan SSB'yi başarılı bir şekilde telafi edip edemeyeceğini araştırıyoruz. Bu amaçla, uyarı-geribildirim etkileşiminden kaynaklanan SSB'den esas olarak etkilenen geribildirim sınıflandırıcısı F_t 'yi ele alıyoruz ve koşullu çıkarım ağaçlarına dayalı RF'lerin ağırlığa duyarlı bir uygulamasını kullanıyoruz [40]. Önem ağırlıklandırma [32], [69], [70] yeniden ağırlıklandırmadan oluşur-

F_t adresindeki her bir eğitim örneğinde aşağıdaki ağırlık kullanılır:

$$w = \frac{P(\mathbf{s} = 1)}{P(\mathbf{s} = 1|x, y)} \tag{10}$$

Burada **s,** her bir örnekle ilişkilendirilen bir seçim değişkenidir T'de$_t$ işlem F_t içindeyse 1, aksi takdirde 0 değerini alır. Böylece, $P(\mathbf{s} = 1|x, y)$, bir örneğin (x, y) F_t eğitim kümesinde olma olasılığına karşılık gelir. (10)'daki ağırlıkların tanımı Bayes teoreminden ve $P(x, \mathbf{y}|\mathbf{s} = 1)$ yanlı birleşik dağılımına göre yansız birleşik dağılım $P(x, y)$'yi ifade etmenin mümkün olduğu gerçeğinden ([19]'da olduğu gibi) kaynaklanmaktadır:

$$P(x, y) = \frac{P(\mathbf{s} = 1)}{P(\mathbf{s} = 1|x, y)} P(x, \mathbf{y}|\mathbf{s} = 1) = wP(x, \mathbf{y}|\mathbf{s} = 1).$$

sınıflandırıcıların performanslarının oldukça dalgalandığını ve Şubat 2015'te bir zirve yaptığını belirtmek gerekir. Bu gerçekten de veri setimizde en fazla sayıda sahte kartın bulunduğu aydır [Şekil 5(b)'de raporlanmıştır]. Buna karşılık, Ekim 2014'te (en düşük sahte kart sayısının görüldüğü dönem)

Tablo V, (10) tarafından sağlanan ağırlıkları kullanarak SSB'yi düzeltirken elde edilen performansı bildirmektedir ve bunların Tablo II'de *F* tarafından elde edilen performanstan daha düşük olduğu ortaya çıkmaktadır. Önem ağırlıklandırması aslında *F*'nin performansını iyileştirmemektedir, bunu uyarı-geribildirim etkileşimi tarafından ortaya çıkarılan SSB'yi telafi ederken bir başarısızlık olarak yorumluyoruz.

(10)'daki $P(\mathbf{s} = 1|x, y)$ ve $P(+|x)$, uyarı-geribildirim etkileşimi nedeniyle yüksek oranda ilişkili olduğundan, önem ağırlığının etkisiz hale geldiğine inanıyoruz. Bu da şu anlama gelmektedir

TABLO V

Q = 15 OLDUĞUNDA F_t İÇİN ORTALAMA P_k , CP_k VE AUC

metric	mean	sd	dataset
P_k	0.68	0.26	2014-2015
P_k	0.59	0.26	2013
CP_k	0.26	0.16	2014-2015
CP_k	0.25	0.13	2013
AUC	0.85	0.06	2014-2015
AUC	0.85	0.06	2013

TABLO VI

ORTALAMA CP_k 30 GÜN KULLANILDIĞINDA (δ = 15, M = 15 VE Q = 30)

classifier	mean	sd	sum of ranks	comparison	dataset
$\mathcal{A}^W$	0.38	0.17	1671.00	a	2014-2015
$\mathcal{F}$	0.36	0.17	1482.50	b	2014-2015
$\mathcal{R}$	0.31	0.17	1234.50	c	2014-2015
$\mathcal{W}$	0.25	0.13	850.50	d	2014-2015
$\mathcal{W}^D$	0.24	0.12	705.50	e	2014-2015
$\mathcal{S}$	0.23	0.12	605.50	f	2014-2015
$\mathcal{A}^W$	0.38	0.14	609.00	a	2013
$\mathcal{F}$	0.35	0.14	541.00	b	2013
$\mathcal{R}$	0.27	0.11	411.50	c	2013
$\mathcal{W}$	0.25	0.13	325.50	d	2013
$\mathcal{W}^D$	0.24	0.12	281.00	e	2013
$\mathcal{S}$	0.20	0.12	198.00	f	2013

Bir işlemin riskli olarak değerlendirilme olasılığı ne kadar yüksekse, $P(\mathbf{s} = 1|x, y)$ olasılığı o kadar büyük ve buna bağlı olarak (10)'daki ağırlığı o kadar düşük olur. Dolayısıyla, önem ağırlıklandırma, geri bildirimler içinde dolandırıcılık olma olasılığı daha yüksek olan örneklerin etkisini azaltır ve bu da uyarı hassasiyetini olumsuz yönde etkiler. Akıl sağlığı kontrolü olarak, bu deneyi son denetimli numunelerin Uyarı-geri bildirim etkileşimi ancak AC500'den daha büyük tutarlı işlemler arasından rastgele seçilir (yukarıdaki deneyle aynı sayı ve sınıf oranlarında). $P(s|y, x) = P(s|x)$ olduğundan, yani x girdisi verildiğinde **s** seçim değişkeni y sınıfından bağımsız olduğundan, SSB'nin bu biçimi ortak değişken kaydırması olarak adlandırılır [42], [59], [68]. durumda, önem ağırlıklandırma bu sapmayı doğru bir şekilde telafi edebilmiştir ve sapması giderilmiş sınıflandırıcı, SSB düzeltilmeden eğitilen benzer bir sınıflandırıcıdan daha iyi performans göstermektedir.

F. Parametrelerin Etkisi

Burada F_t ve A^W $_t$ var dır tarafından etkilenmiştir: 1) sınıflandırıcılarımızı eğitmek için dikkate alınan geri bildirim günlerinin sayısı (yani Q); 2) araştırmacılar tarafından günlük olarak kontrol edilen kart sayısı; ve 3) (9)'daki toplama sınıflandırıcısını düzenleyen α parametresi. Bu amaçla, δ = 15 günlük doğrulama gecikmesini dikkate alıyoruz, öyle ki F_t 30 günlük geri bildirimlerle eğitilmiştir (Q = 30, M = 15, ve δ = 15) ve gecikmeli denetimli örnekler 15 gün sonra gelir. Tablo VI, Q = 30 günlük geri bildirim kullanılarak eğitildiğinde F'nin CP_k açısından Q = 15'e göre daha iyi olduğunu göstermektedir (bkz. Tablo II). Aynı durum, F tarafından elde edilen üstün performansın bir sonucu olarak müfettişlerin 100'den fazla kartı kontrol edebildiğini ve daha fazla kart kontrol edilebildiğinde uyarı hassasiyetini doğru bir şekilde değerlendirmek için sahtekarlık tespit performansını NCP_k açısından raporladığını göstermektedir. Bu sonuç, daha fazla geri bildirime sahip olmanın üstün dolandırıcılık tespit performansını garanti ettiğini doğrulamaktadır. Bu analiz, daha fazla araştırmacı istihdam etmenin maliyetinin dolandırıcılık tespit performansında beklenen iyileşme ile telafi edilip edilmeyeceğine karar vermek zorunda olan şirketler için bir kılavuz olarak düşünülebilir.

Öğrenme stratejimizdeki bir diğer önemli parametre de (9)'daki geri beslemeli ve gecikmeli sınıflandırıcıların katkısını dengeleyen α'dır. Bu parametre, günlük bazda uyarlanabilir hale getirmek için birden fazla strateji araştırıldıktan sonra deneysel olarak 0,5 olarak ayarlanmıştır. Fikrimiz, t -1 günü boyunca F_{t-1} ve D_{t-1} tarafından elde edilen kesinliği (veya diğer performans ölçütlerini) dikkate almak ve ardından F_t ve D 'ye ağırlıklar atamaktı.
D_t buna göre (sınıflandırıcı t - 1 günü boyunca en iyisiydi, t günü boyunca ağırlık ne kadar büyükse). Ne yazık ki, hiçbiri Uygulanan çözümlerin iki öncülün ortalamasından daha iyi performans gösterdiği görülmüştür, yani α_t = 0,5 $\forall t$.

Bu nedenle, kayan pencere çözümü üzerinde kapsamlı bir simülasyon gerçekleştirdik ve burada her gün $\alpha_t \in \{0,1, 0,2, \ldots 0.9\}$ *değerlerini test* ettiğimiz kapsamlı bir simülasyon gerçekleştirdik ve ardından P_k açısından en iyi performansı gösteren toplamayı sağlayan α^* değerini seçtik. Böyle bir optimum ağırlık seçimi elbette gerçek dünyadaki bir FDS'de uygulanabilir değildir,
çünkü her $\alpha_t \in \{0.1, 0.2, \ldots$ için geri bildirim talep etmek gerekecektir. , 0.9}. Bununla birlikte, her gün α^* *ayarı,* α_t = 0,5 $\forall t$ ayarına göre minimum iyileştirme sağlamıştır. Bu durum, α^* değerinin 0,5 civarında zirveli bir $_t$ dağılıma sahip olması ve ortalama$(\alpha^*) \approx 0,52$ olmasıyla açıklanabilir. P_k değeri α = 0.1 ve α = 0.9'a yaklaştıkça sürekli olarak azalmaktadır, bu da α'nın uç değerlerinin nadiren en iyi seçenek olduğunu göstermektedir. Bu uç durumlarda, A_t aşağıdaki seçeneklerden birine yaklaşır
D_t veya F_t (ki bunların en iyi seçenekler olmadığı gösterilmiştir) ve en düşük ağırlığı alan sınıflandırıcının performansını iyileştirmek ve ağırlığını artırmak için geri bildirim talep etme şansı çok azdır.

VII. SONUÇ

Dolandırıcılık tespit problemini ele alan çalışmaların çoğu kredi kartı işlemlerinde (bkz. [5], [23], [63]) gerçekçi olmayan

A^W için de geçerlidir.
bu durumda doğrulama gecikmesindeki artış.

Bu parametrenin F ve A'nın performansını nasıl etkilediğini göstermek için bu deneyi günlük daha fazla sayıda geri bildirimi dikkate alarak tekrarlıyoruz^W .

sınıflandırıcıyı eğitmek için her işlemin sınıfının hemen sağlandığını varsayar. Burada FDS'nin gerçek dünyadaki çalışma koşullarını ayrıntılı olarak analiz ediyor ve ilgili eklemli sınıflandırma probleminin resmi bir tanımını sunuyoruz. Özellikle, sınıflandırıcıyı eğitmek/güncellemek için son denetimli örnekleri sağlayan mekanizma olan uyarı-geri bildirim etkileşimini tanımladık. Ayrıca, literatürde kullanılan geleneksel performans ölçütlerinin aksine, gerçek dünyadaki bir FDS'de, araştırmacılar yalnızca birkaç uyarıyı kontrol edebildiğinden, bildirilen uyarıların kesinliğinin muhtemelen en anlamlı olanı olduğunu iddia ediyoruz.

Gerçek dünya işlemlerinden oluşan iki geniş veri seti üzerinde yaptığımız deneyler, kesin uyarılar elde etmek için öğrenme problemi sırasında geri bildirimlere daha fazla önem vermenin zorunlu olduğunu göstermektedir. Şaşırtıcı olmayan bir şekilde, geri bildirimler önerilen öğrenme stratejisinde merkezi bir rol oynamaktadır.

geri bildirimler üzerine bir sınıflandırıcı ve gecikmeli denetimli örnekler üzerine bir sınıflandırıcı eğitmek ve ardından uyarıları belirlemek için son değerlerini toplamak. Deneylerimiz ayrıca, öğrenme sürecinde geri bildirimlerin etkisini azaltan çözümlerin (örneğin, geri bildirimleri ve gecikmeli denetimli örnekleri karıştıran veya örnek ağırlıklandırma şemaları uygulayan sınıflandırıcılar) genellikle daha az kesin uyarılar verdiğini göstermektedir.

Gelecekteki çalışmalar, geri bildirimler ve gecikmeli denetimli örnekler üzerinde eğitilen sınıflandırıcılar için uyarlanabilir ve muhtemelen doğrusal olmayan toplama yöntemlerinin incelenmesiyle ilgilidir. Ayrıca, sonsal olasılıkların doğrusal toplanmasının yerini almak üzere özel olarak tasarlanacak bir *sıralamayı öğrenme* yaklaşımı [46] uygulayarak uyarı hassasiyetini daha da artırmayı umuyoruz. Son olarak, çok umut verici bir araştırma yönü, öğrenme sürecinde az sayıda yeni etiketsiz işlemden de yararlanmak için yarı denetimli öğrenme yöntemleriyle [16], [39] ilgilidir.

REFERANSLA

R

[1] E. Aleskerov, B. Freisleben ve B. Rao, "CARDWATCH: Kredi kartı dolandırıcılığı tespiti için sinir ağı tabanlı bir veritabanı madenciliği sistemi," *Proc. IEEE/IAFE Computat. Intell. Financial Eng.*, Mart 1997, s. 220-226.

[2] C. Alippi, G. Boracchi ve M. Roveri, "A just-in-time adaptive classification system based on the intersection of confidence intervals rule," *Neural Netw.,* vol. 24, no. 8, pp. 791-800, 2011.

[3] C. Alippi, G. Boracchi ve M. Roveri, "Hiyerarşik değişim algılama testleri," *IEEE Trans. Neural Netw. Learn. Syst.*, cilt 28, no. 2, pp. 246-258, Şubat 2016.

[4] C. Alippi, G. Boracchi, ve M. Roveri, "Just-in-time classifiers for recurrent concepts," *IEEE Trans. Neural Netw. Learn. Syst.,* vol. 24, no. 4, pp. 620-634, Nisan 2013.

[5] B. Baesens, V. Van Vlasselaer ve W. Verbeke, *Tanımlayıcı, Tahmine Dayalı ve Sosyal Ağ Teknikleri Kullanarak Dolandırıcılık Analitiği: Dolandırıcılık Tespiti için Veri Bilimi Kılavuzu*. Hoboken, NJ, ABD: Wiley, 2015.

[6] A. C. Bahnsen, D. Aouada, and B. Ottersten, "Example-dependent cost-sensitive decision trees," *Expert Syst. Appl.*, cilt 42, no. 19, pp. 6609-6619, 2015.

[7] A. C. Bahnsen, D. Aouada, A. Stojanovic, ve B. Ottersten, "Detecting credit card fraud using periodic features," in *Proc. 14th Int. Konf. Mach. Learn. Appl.*, Aralık 2015, s. 208-213.

[8] S. Bhattacharyya, S. Jha, K. Tharakunnel ve J. C. Westland, "Kredi kartı dolandırıcılığı için veri madenciliği: A comparative study," *Decision Support Syst.,* vol. 50, no. 3, pp. 602-613, 2011.

[9] A. Bifet ve R. Gavalda, "Learning from time-changing data with adaptive windowing," in *Proc. SDM,* vol. 7. 2007, pp. 443-448.

[10] R. Bolton ve D. Hand, "İstatistiksel sahtekarlık tespiti: Bir inceleme," *Stat. Sci.*, vol. 17, no. 3, pp. 235-249, 2002.

[11] R. J. Bolton ve D. J. Hand, "Sahtekarlık tespiti için denetimsiz profilleme yöntemleri," *Kredi Puanlaması Kredi Kontrolü VII*. Londra, Birleşik Krallık: Imperial College London, 2001, s. 235-255.

[12] R. Brause, T. Langsdorf ve M. Hepp, "Neural data mining for credit card fraud detection," in *Proc. Tools Artif. Intell.*, Temmuz 1999, s. 103-106.

[13] L. Breiman, "Random forests," *Mach. Learn.*, cilt 45, no. 1, s. 5-32, 2001.

[14] M. Carminati, R. Caron, F. Maggi, I. Epifani ve S. Zanero, *BankSealer: Çevrimiçi Bankacılık Dolandırıcılık Analizi ve Soruşturması için Bir Karar Destek Sistemi*, Berlin, Almanya: Springer, 2014, s. 380-394.

[15] P. K. Chan, W. Fan, A. L. Prodromidis, ve S. J. Stolfo, "Distributed data mining in credit card fraud detection," *IEEE Intell. Syst. Uygulamaları*, cilt 14, no. 6, s. 67-74, Kasım 1999.

[16] O. Chapelle, B. Scholkopf ve A. Zien, "Semi-supervised learning," *IEEE Trans. Neural Netw.*, vol. 20, no. 3, pp. 542-542, 2009.

[17] N. Chawla, K. Bowyer, L. O. Hall ve W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, cilt 16, s. 321-357, 2002.

[18] S. Chen ve H. He, "Durağan olmayan dengesiz veri akışının artımlı öğrenmesine doğru: A multiple selectively recursive approach," *Evolving Syst.,* vol. 2, no. 1, pp. 35-50, 2011.

[19] C. Cortes, M. Mohri, M. Riley ve A. Rostamizadeh, "Sample selec- tion bias correction theory," in *Algorithmic Learning Theory*. Berlin, Almanya: Springer, 2008, s. 38-53.

[20] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi ve G. Bontempi, "Kredi kartı dolandırıcılığı tespiti ve gecikmeli denetimli bilgi ile kavram kayması adaptasyonu," *Proc. Joint Conf. Neural Netw.*, 2015, s. 1-8.

[21] A. Dal Pozzolo, O. Caelen ve G. Bontempi, "*Dengesiz* sınıflandırma görevlerinde alt örnekleme ne zaman etkilidir?" in *Machine Learning and Knowledge Discovery in Databases*. Cambridge, U.K.: Springer, 2015.

[22] A. Dal Pozzolo, O. Caelen, R. A. Johnson ve G. Bontempi, "Calibrating probability with undersampling for unbalanced classification," in *Proc. IEEE Symp. Ser. Computat. Intell. içinde,* Aralık 2015, s. 159-166.

[23] A. Dal Pozzolo, O. Caelen, Y.-A. L. Borgne, S. Waterschoot, ve G. Bontempi, "Bir uygulayıcı perspektifinden kredi kartı dolandırıcılığı tespitinde öğrenilen dersler," *Expert Syst. Appl.*, cilt 41, no. 10, pp. 4915-4928, 2014.

[24] A. Dal Pozzolo, R. A. Johnson, O. Caelen, S. Waterschoot, N. V. Chawla ve G. Bontempi, "Using HDDT to avoid instances propagation in unbalanced and evolving data streams," in *Proc. Joint Conf. Neural Netw.*, 2014, s. 588-594.

[25] J. Demšar, "Birden fazla veri seti üzerinde sınıflandırıcıların istatistiksel karşılaştırmaları," *J. Mach. Learn. Res.*, cilt 7, s. 1-30, Ocak 2006.

[26] G. Ditzler ve R. Polikar, "Incremental learning of concept drift from streaming imbalanced data," *IEEE Trans. Knowl. Data Eng.,* vol. 25, no. 10, pp. 2283-2301, Ekim 2013.

[27] G. Ditzler, M. Roveri, C. Alippi, and R. Polikar, "Learning in nonstationary environments: A survey," *IEEE Comput. Intell. Mag.*, vol. 10, no. 4, pp. 12-25, Nisan 2015.

[28] J. R. Dorronsoro, F. Ginel, C. Sgnchez, ve C. S. Cruz, "Kredi kartı işlemlerinde nöral dolandırıcılık tespiti," *IEEE Trans. Neural Netw.*, vol. 8, no. 4, s. 827-834, Temmuz 1997.

[29] C. Elkan, "The foundations of cost-sensitive learning," in *Proc. Int. Joint Conf. Artif. Intell.*, 2001, cilt 17. no. 1, s. 973-978.

[30] R. Elwell ve R. Polikar, "Incremental learning of concept drift in nonstationary environments," *Trans. Neural Netw.*, cilt 22, no. 10, pp. 1517-1531, 2011.

[31] W. Fan ve I. Davidson, "On sample selection bias and its efficient correction via model averaging and unlabeled examples," in *Proc. SDM*, 2007, pp. 320-331.

[32] W. Fan, I. Davidson, B. Zadrozny, and P. S. Yu, "An improved categorization of classifier's sensitivity on sample selection bias," in *Proc. 5th Int. Konf. Veri Madenciliği,* Kasım 2005, s. 4.

[33] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *J. Amer. Stat. Assoc.*, cilt 32, no. 200, s. 675-701, 1937.

[34] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," in *Advances in Artificial Intelligence*. Berlin, Almanya: Springer, 2004, s. 286-295.

[35] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, ve A. Bouchachia, "A survey on concept drift adaptation," *ACM Comput. Surv.,* cilt 46, no. 4, p. 44, 2014.

[36] J. Gao, B. Ding, W. Fan, J. Han ve P. S. Yu, "Classifying data streams with skewed class distributions and concept drifts," *IEEE Internet Comput.,* vol. 12, no. 6, pp. 37-49, Nov. 2008.

[37] D. Hand, "Sınıflandırıcı performansının ölçülmesi: ROC eğrisi altındaki alana tutarlı bir alternatif," *Mach. Learn.*, cilt 77, no. 1, pp. 103-123, 2009.

[38] H. He ve E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.,* vol. 21, no. 9, pp. 1263-1284, Eylül 2009.

[39] M. J. Hosseini, A. Gholipour, and H. Beigy, "An ensemble of cluster-based classifiers for semi-supervised classification of non-stationary data streams," *Knowl. Inf. Syst.*, vol. 46, no. 3, pp. 567-597, 2016.

[40] T. Hothorn, P . Bühlmann, S . Dudoit, A . Molinaro ve M. J. van der Laan, "Survival ensembles," *Biostatistics*, vol. 7, no. 3, pp. 355-373, 2006.

[41] S. Jha, M. Guillen, and J. C. Westland, "Employing transaction aggregation strategy to detect credit card fraud," *Expert Syst. Appl.*, vol. 39, no. 16, pp. 12650-12657, 2012.

[42] M. G. Kelly, D. J. Hand ve N. M. Adams, "The impact of changing populations on classifier performance," in *Proc. 25th Int. Konf. Knowl. Discovery Data Mining,* 1999, s. 367-371.

[43] J. Z. Kolter ve M. A. Maloof, "Dinamik ağırlıklı çoğunluk: An ensemble method for drifting concepts," *J. Mach. Learn. Res.*, cilt 8, pp. 2755-2790, Aralık 2007.

[44] G. Krempl ve V. Hofer, "Classification in presence of drift and latency," in *Proc. 11th Data Mining Workshops*, Aralık 2011, s. 596-603.

[45] M. Krivko, "A hybrid model for plastic card fraud detection systems," *Expert Syst. Appl.*, vol. 37, no. 8, pp. 6070-6076, 2010.
[46] T.-Y. Liu, "Learning to rank for information retrieval," *Found. Trends Inf. Retr.*, vol. 3, no. 3, pp. 225-331, 2009.
[47] N. Mahmoudi ve E. Duman, "Detecting credit card fraud by mod- ified fisher discriminant analysis," *Expert Syst. Appl.*, vol. 42, no. 5, pp. 2510-2516, 2015.
[48] H. B. Mann ve D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *Ann. Math. Statist.,* vol. 18, no. 1, pp. 50-60, 1947.
[49] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla ve F. Herrera, "A unifying view on dataset shift in classification," *Pattern Recognit.,* vol. 45, no. 1, pp. 521-530, 2012.
[50] K. Nishida ve K. Yamauchi, "Detecting concept drift using statisti- cal testing," in *Discovery Science*. Berlin, Almanya: Springer, 2007, s. 264-269.
[51] D. Olszewski, "Fraud detection using self-organizing map visualizing the user profiles," *Knowl.-Based Syst.,* vol. 70, pp. 324-334, Nov. 2014.
[52] C. Phua, V. Lee, K. Smith ve R. Gayler. (Eylül 2010). "A comprehen- sive survey of data mining-based fraud detection research." [Çevrimiçi]. Mevcut: https://arxiv.org/abs/1009.6119
[53] J. Plasse ve N. Adams, "Handling delayed labels in temporally evolving data streams," in *Proc. Int. Conf. Big Data,* 2016, s. 2416-2424.
[54] J. T. Quah ve M. Sriganesh, "Real-time credit card fraud detection using computational intelligence," *Expert Syst. Appl.*, vol. 35, no. 4, pp. 1721-1732, 2008.
[55] Y. Sahin, S. Bulkan, and E. Duman, "A cost-sensitive decision tree approach for fraud detection," *Expert Syst. Appl.*, cilt 40, no. 15, pp. 5916-5923, 2013.
[56] D. Sánchez, M. A. Vila, L. Cerda, and J. M. Serrano, "Association rules applied to credit card fraud detection," *Expert Syst. Appl.*, cilt 36, no. 2, pp. 3630-3640, 2009.
[57] J. C. Schlimmer ve R. H. Granger, Jr., "Incremental learning from noisy data," *Mach. Learn.*, vol. 1, no. 3, pp. 317-354, 1986.
[58] B. Settles, "Active learning literature survey," *Univ. Wisconsin, Madison,* vol. 52, nos. 55-66, p. 11, 2010.
[59] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *J. Stat. Planning Inference*, vol. 90, no. 2, pp. 227-244, 2000.
[60] P. Sobhani ve H. Beigy, *Veri Akışları için Yeni Sürüklenme Tespit Yöntemi*. Berlin, Almanya: Springer, 2011.
[61] W. N. Street ve Y. Kim, "A streaming ensemble algorithm (SEA) for large-scale classification," in *Proc. 7th Int. Konf. Knowl. Discovery Data Mining,* 2001, s. 377-382.
[62] D. K. Tasoulis, N. M. Adams ve D. J. Hand, "Unsupervised clustering in streaming data," in *Proc. Int. Konf. Veri Madenciliği Çalıştayları*, 2006, pp. 638-642.
[63] V. Van Vlasselaer ve *diğerleri,* "*APATE*: A novel approach for automated credit card transaction fraud detection using network-based extensions," *Decision Support Syst.,* vol. 75, pp. 38-48, Jul. 2015.
[64] S. Wang, L. L. Minku ve X. Yao, "Çevrimiçi sınıf dengesizliği öğrenimi için yeniden örnekleme tabanlı topluluk yöntemleri," *Trans. Knowl. Data Eng.*, vol. 27, no. 5, pp. 1356-1368, Mayıs 2015.
[65] D. J. Weston, D. J. Hand, N. M. Adams, C. Whitrow, and P. Juszczak, "Plastic card fraud detection using peer group analysis," *Adv. Veri Analizi. Classification*, vol. 2, no. 1, pp. 45-62, 2008.
[66] C. Whitrow, D. J. Hand, P. Juszczak, D. J. Weston ve N. M. Adams, "Transaction aggregation as a strategy for credit card fraud detec- tion," *Data Mining Knowl. Discovery*, cilt 18, no. 1, s. 30-55, 2009.
[67] G. Widmer ve M. Kubat, "Learning in the presence of concept drift and hidden contexts," *Mach. Learn.*, cilt 23, no. 1, s. 69-101, 1996.
[68] K. Yamazaki, M. Kawanabe, S. Watanabe, M. Sugiyama, ve K.-R. Müller, "Asymptotic Bayesian generalization error when training and test distributions are different," in *Proc. 24th Int. Konf. Mach. Learn.*, 2007, s. 1079-1086.
[69] B. Zadrozny, "Learning and evaluating classifiers under sample selection bias," in *Proc. 21st Int. Konf. Mach. Learn.*, 2004, s. 114.
[70] B. Zadrozny, J. Langford ve N. Abe, "Cost-sensitive learning by cost-proportionate example weighting," in *Proc. Int. Conf. Veri Madenciliği*, Kasım 2003, s. 435-442.
[71] V. Zaslavsky ve A. Strizhak, "Kendi kendini organize eden haritalar kullanarak kredi kartı dolandırıcılığı tespiti," *Inf. Secur.*, cilt 18, s. 48, 2006.
[72] I. Žliobaitė. (Ekim 2010). "Kavram kayması altında öğrenme: Genel bir bakış." [Çevrimiçi]. Mevcut: https://arxiv.org/abs/1010.4784

Andrea Dal Pozzolo, 2011 yılında Univer- sitá di Bologna, Bologna, İtalya, İstatistik Fakültesi'nden yüksek lisans derecesini *(cum laude) ve* 2015 yılında Université Libre de Bruxelles, Brüksel, Belçika, Makine Öğrenimi Grubu'ndan doktora derecesini aldı ve burada dolandırıcılık tespiti için makine öğrenimi ve istatistiksel teknikler üzerine çalıştı.

Halen büyük bankalar ve sigorta şirketleri için danışmanlık yapmaktadır. Güncel araştırma alanları arasında dengesiz veri akışları, maliyete duyarlı öğrenme ve kavram kayması yer almaktadır.

Giacomo Boracchi, 2004 yılında Università degli Studi di Milano, Milano, İtalya'dan matematik alanında yüksek lisans derecesini ve 2008 yılında Politecnico di Milano, Milano'dan bilgi teknolojileri alanında doktora derecesini almıştır.

2004-2005 yıllarında Tampere Inter- national Center for Signal Processing, Tampere, Finlandiya'da araştırmacı olarak çalışmıştır. Halen Milano Politecnico di Milano Dipartimento di Elettronica, Informazione e Bioingegneria'da Yardımcı Doçent olarak görev yapmaktadır. Şu anki araştırma alanları şunlardır Durağan olmayan ortamlar için öğrenme yöntemlerinin yanı sıra görüntü işleme ve analizi için matematiksel ve istatistiksel yöntemler.

Dr. Boracchi 2015 yılında IBM Fakülte Ödülü ve IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS ödülünü almıştır. 2016'da Üstün Makale Ödülü. 2017'de Nokia Misafir Profesör Bursu'nu aldı.

Olivier Caelen, doktora derecesini Profesör G. Bontempi'nin danışmanlığında Université Libre de Bruxelles, Brüksel, Belçika'daki Makine Öğrenimi Grubu'ndan almıştır.

Kredi kartı dolandırıcılığı tespit ekibinde beş yıl geçirdikten sonra şu anda Worldline (bir Atos şirketi), Brüksel, Belçika'da Ar-Ge Yüksek İşlem ve Hacim Ekibinde çalışmaktadır. Mevcut araştırma alanları arasında anomali ve dolandırıcılık tespiti için makine öğrenimi teknikleri yer almaktadır.

Cesare Alippi (SM'94-F'06) halen Politecnico di Milano, Milano, İtalya ve Università della Svizzera Italiana, Lugano, İsviçre'de profesör olarak görev yapmaktadır. Mevcut araştırma alanları arasında durağan olmayan ortamlarda adaptasyon ve öğrenme ile gömülü ve siber fiziksel sistemler için zeka yer almaktadır.

Dr. Alippi, IEEE Computational Intelligence Society Yönetim Komitesi, International Neural Network Society Yönetim Kurulu ve Neural Network Society Yönetim Kurulu üyesidir. Avrupa Sinir Ağları Topluluğu'nun üyesidir. Uluslararası Sinir Ağları Topluluğu'ndan Gabor Ödülü ve 2016 yılında IEEE Com- putational Intelligence Society Outstanding TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS Paper Award ve 2004 yılında IEEE Instrumentation and Measurement Society Young Engineer Award ödüllerini almıştır.

Gianluca Bontempi (SM'12), Milano Politecnico, Milano, İtalya'dan elektronik mühendisliği alanında yüksek lisans derecesi (Hons.) ve Université Libre de Bruxelles (ULB), Brüksel, Belçika'dan uygulamalı bilimler alanında doktora derecesi almıştır.

Halen ULB Bilgisayar Bilimleri Bölümü'nde Profesör olarak görev yapmakta ve ULB Makine Öğrenimi Grubu'nun Eş Başkanı ve Brüksel'deki Üniversitelerarası Biyoinformatik Enstitüsü'nün Direktörü olarak çalışmaktadır. Akademi ve özel sektörde araştırma projelerinde yer almıştır. Avrupa'nın dört bir yanındaki şirketler. İki uluslararası yarışmada ödül alan 200'den fazla bilimsel yayının ve veri madenciliği ve tahmin yazılımının yazarlığını veya ortak yazarlığını yapmıştır. Şu anki araştırma alanları arasında veri madenciliği, ölçeklenebilir makine öğrenimi, biyoinformatik ve zaman serisi tahmini bulunmaktadır.