

# Müşteri Kayıp Analizi (Customer Churn Analysis)

Cansu Yalçın<sup>a</sup>

<sup>1</sup> Antalya Bilim University, Computer Engineering, Turkey

E-mail: cansuyalcin.00@gmail.com

## Özet

Firmalar için mevcut müşterilerinin kaybedilmemesi ve müşteri kayıplarının en aza indirmesi önem arz etmektedir, bu amaca yönelik olarak müşteri kayıp analizleri yaygın olarak kullanılmaya başlamıştır. Özellikle üyelik sistemi ile çalışan bankacılık, telekom ve sigorta gibi sektörlerde müşteri kaybını en aza indirmek ve bu kayıpları önceden tespit edebilmek, daha sonraki nakit akışı, müşteri ilişkileri ve pazarlama hizmetleri gibi konular açısından çok önemlidir. Bu yazının amacı, bankacılık sektörüne ait müşteri verisi üzerinden makine öğrenmesi algoritmaları kullanılarak müşterilerin kayıp analizinin yapılmasıdır. Çalışma kapsamında, farklı makine öğrenme algoritmaları kullanılarak kodlamasız yaklaşım felsefesinin savunuculuğunu yapan, açık kaynak kodlu, görsel bir veri bilimi geliştirme ortamı olan Knime, yaygın olarak kullanılan Python kodlama ortamında SKLearn kütüphanesi ve AutoML felsefesini benimsemiş olan OptiScorer ürünleri kullanılarak karşılaştırılmış, %90'a kadar başarı elde edilmiştir.

**Anahtar Kelimeler:** Müşteri İlişkileri yönetimi, Makine öğrenmesi, Veri Bilimi.

## Abstract

Companies should minimize customer losses and analyze customer losses well in order to prevent their existing customers from moving to a competing firm. Especially in banking, telecom and insurance sectors working with membership system, it is crucial to minimize customer loss and to detect these losses in advance. The purpose of this paper is to analyze the loss of customers by using machine learning algorithms on customer data of banking sector. In this study Knime, an open source, visual data science development environment that advocates a non-coding approach using different machine learning algorithms, and SKLearn library in the widely used Python coding environment, OptiScorer products which adopt AutoML philosophy are used and results are compared. Up to %90 success was achieved.

**Keywords:** Customer Relationship management, Machine learning, Data Science.

## 1. Giriş

Müşteri kaybı analizi, ayrılmayı düşünen potansiyel abonelerin tahminini yapan ve onları iptalden uzaklaştırmanın yollarını arayan bir süreç olarak tanımlanabilir. Bu tahminler sayesinde müşterinin kaybının önüne geçilmesi için çözümler üretilebilmektedir. Yeni bir müşteri edinme maliyeti, mevcut müşterileri elde tutma maliyetinin yaklaşık 5 katı, memnun olmayan müşteriyi geri kazanmanın maliyeti mevcut müşteriyi elde tutmanın yaklaşık 10 katıdır [1]. Bu nedenle şirketler, mevcut müşterilerini elde tutmaya yönelik bir takım sadakat programları geliştirirler. Müşteri kaybının en yoğun ve hızlı yaşandığı sektörlerden biri olarak düşünülen bankacılık sektöründe de müşteriyi elde tutmak büyük bir avantajdır. Müşterileri elinde tutma, sadece en iyi ve faydalı müşterileri rakiplerden korumak değil, aynı zamanda onlar arasındaki ilişki ve bağlantıları sürdürmek ve böylece hizmet kullanımını ve buna bağlı geliri korumaktır [2].

Ayrıca bilgisayar teknolojilerindeki gelişmeler, işletmelerin çok miktarda veriyi saklayabilmesini ve işleyerek anlamlı bilgilere dönüştürmesini mümkün hale getirmiştir. Bugün, işletmeler satın alma davranışından demografik özelliklere kadar müşterilerle ilgili pek çok detayı veri tabanlarında tutmaktadır. Veri madenciliği sonuçları müşteri kaybının engellenmesi için kullanılabilir.

Makalenin 2. bölümünde veri madenciliğinden ve kayıp analizinden bahsedilmiş, 3. bölümde Endüstriler arası standart işleme – Veri bilimi (Cross Industry Standard Processing – Data Mining) CRISP-DM adımları verinin üzerinde uygulanmış ve son bölüm olan 4. bölümde ise Knime, Python ve OptiScorer gibi farklı platformlarda ve farklı algoritmalara kullanılarak elde edilen sonuçlar karşılaştırılmış ve çıkarımlarda bulunulmuştur.

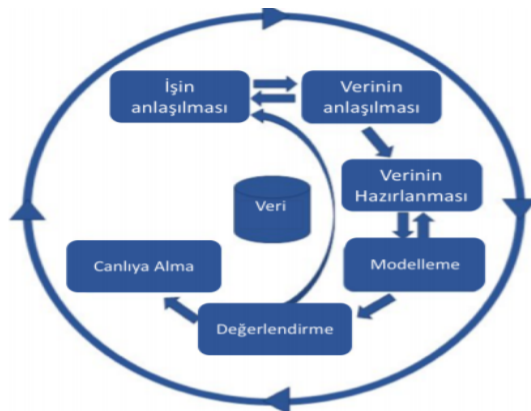
## 2. Veri Madenciliği ve Kayıp Analizi ( Data Mining and Churn Analysis )

Veri madenciliği, önceden bilinmeyen ilişki ve trendlerin bulunması için bugünün endüstrisinde yaratılan büyük miktarlardaki veriyi analiz eden bir yoldur [3]. İşletmeler için önemli değerlerden birisi de kayıpların önceden doğru şekilde tahmin edilebilmesidir. Çoğu iş zekası uygulaması [4], bu amaca yönelik olarak veri madenciliğine dayalı [5] uygulamalar geliştirmektedir. Bu uygulamalardan biri olarak kayıp analizi düşünebilir.

Kayıp analizinde daha verimli sonuçlara ulaşılabilmek adına bu çalışmamızda veri madenciliğinden faydalanılmış, çalışmanın amacına uygun olarak ayrılma potansiyeli olan müşteriler makine öğrenmesi algoritmaları kullanılarak sınıflandırılmıştır.

### 2.1 Metodoloji

CRIPS-DM veri madenciliği projelerini planlama ve yürütmede kullanılan bir süreç modelidir.



Şekil 1: CRISP-DM Adımları ve Akışı

Veri madenciliği sürecinde çalışmamızda CRISP-DM metodolojisi kullanılmış ve takip eden adımlar izlenmiştir.

1. İşi Tanımlama ( Business Understanding )
2. Veriyi Anlama ( Data Understanding )
3. Veriyi Hazırlama ( Data Preparation )
4. Modelleme ( Modeling )
5. Değerlendirme ( Evaluation )
6. Uygulama ( Deployment )

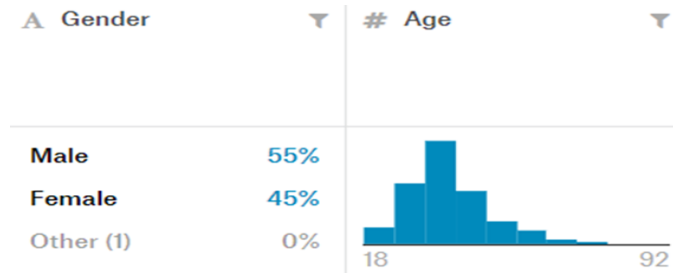
### 3. CRISP-DM Metodolojisinin Banka Müşteri Verisine Uygulanması

#### 3.1 İşi Tanımlama ( Business Understanding )

Bir bankanın ayrılma potansiyeline sahip olan müşterilerinin analizini yapabilmek ve sonrasında bu potansiyele sahip olan müşteriler için sadakat programları geliştirilmesi hedeflenmektedir. Sadakat programları müşterilerin şirketin ürün veya hizmetlerini kullandıkça ödüllendirilmeleri suretiyle müşteri deneyimine katkı yapmayı ve dolayısıyla müşterinin marka ile olan yolculuğunun kesintisiz devam etmesini sağlamayı hedeflemektedir [6].

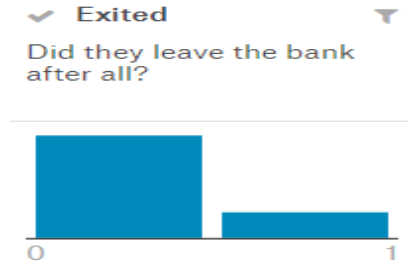
#### 3.2 Veriyi Anlama ( Data Understanding )

Bankanın müşteri verisine Kaggle.com sitesinden ulaşılmıştır. Bu veri kümesi Müşteri ID, Kredi skoru, Müşterinin bankaya üyelik süresi, bankada kaç adet hesabının bulunduğu, kredi kartına sahip olup olmadığı bilgisi, maaş bilgisi, bulunduğu ülke, cinsiyet ve yaş gibi bilgiler içeren 14 farklı kolondan ve 10.000 satırdan oluşmaktadır. Şekil 2’de verinin cinsiyet ve yaş dağılımı gösterilmiştir.



Şekil 2: Verinin cinsiyet ve yaş dağılımı

Veri kategorik ve sayısal verileri beraber bulundurduğu için özellikle seçilmiştir. Veri müşterinin bankadan ayrılıp ayrılmayacağına bilgisini ikili ( Binary ) şekilde belirttiği için yüzde olarak hangi sınıfa dahil olacağına bilgisi skorlama yapılarak bulunmuştur. Şekil 3’de ikili ( Binary ) müşterinin ayrılıp ayrılmama bilgisi gösterilmiştir.



Şekil 3: İkili ( Binary ) müşterinin ayrılıp ayrılmama bilgisi

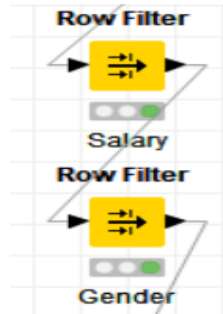
### 3.3. Veriyi Hazırlama ( Data Preparation )

Veriyi hazırlama aşamaları bu çalışmada veri temizleme ve veri dönüşümü olmak üzere iki adettir.

#### 3.3.1 Veri Temizleme

Veri temizleme adımı veri madenciliği sürecinin en kritik ve en çok zaman alan adımıdır [7]. Veri temizleme esnasında Knime Analytics Platform [8] kullanılarak kolonlardan eksik veriler temizlenmiş ve bir sonraki adım olan veri dönüşümüne hazır hale getirilmiştir.

Şekil 4’te satır filtreleri ( Row Filter ) kullanılarak Maaş ( Salary ) ve Cinsiyet ( Gender ) kolonlarına ait olan eksik verilerin temizlenmesi gösterilmektedir. Satır filtresinde sadece eksik verilerin ait olduğu satırlar temizlenmiştir. Eksik verilerin temizlenmesi sonraki adımlarda algoritmaların düzgün çalışması için önemlidir.



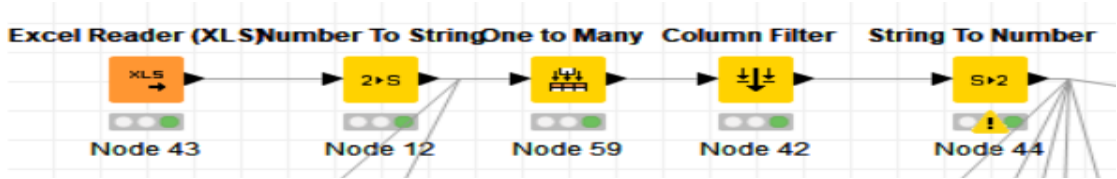
Şekil 4: Knime Analytics Platform, Veri Temizleme ( Data Cleaning )

#### 3.3.2 Veri Dönüşümü

Veri dönüşümü adımı özellikle ilerleyen adımlarda uygulanacak algoritmaların problemsiz bir şekilde çalışması için çok önemlidir. Çalışmamızda da veride yer alan Coğrafya ( Geography ) ve Cinsiyet ( Gender ) kolonlarının kategorik verileri dönüşümü Knime Analytics Platform ve Spyder (Python 3.7) kullanılarak gerçekleştirilmiştir.

Şekilde 5’de Excel Reader düğümü kullanılarak excel formatında olan veri Knime Platformuna aktarılır. Sayıdan diziye çevirme düğümü ( Number to String ) ikili tam sayı ( binary integer ) olarak verilen müşteri ID, kredi kartı olup olmadığı, aktif üye olup olmadığı ve bankadan ayrılıp ayrılmadığı bilgisini dizi ( string ) haline getirmiştir. Bire çok düğümü ( One to Many ) 3 farklı string değerden oluşan Coğrafya ( Geography ) kolonunu 3 farklı ikili ( binary ) kolon haline, 2 farklı string değerden oluşan Cinsiyet ( Gender ) kolonunu 2 farklı ikili ( binary ) kolon haline böler.

Kolon filtresi 3 farklı string değerden oluşan Coğrafya ( Geography ) kolonunu ve 2 farklı string değerden oluşan Cinsiyet (Gender) kolonunu temizler. Diziden sayıya çevirme düğümü ( String to Number ) dizi olarak verilmiş olan Denge ( Balance ) ve Maaş ( Salary ) kolonlarını tam sayıya ( integer ) çevirir. Bu veri dönüştürme işlemleri uygulanacak algoritmaların düzgün sonuçlar vermesine yardımcı olur.



Şekil 5: Knime Analytics Platform, Veri Ön İşleme (Data Preprocessing)

Kod 1’de Spyder (Python 3.7) ile hazırlanmış olan veri ön işleme (Data Preprocessing) adımını görmekteyiz. Coğrafya ( Geography ) ve Cinsiyet (Gender) kolonları ayrılır ve SKLearn kütüphanesi altından LabelEncoder [9] kullanılır. LabelEncoder her bir Coğrafya ( Geography ) değeri için sıfırdan başlayarak sıralı farklı değerler verecektir. OneHotEncoder [10] Coğrafya ( Geography ) ve Cinsiyet (Gender) kolonlarında her bir değer için binary kolon oluşturacaktır.

```

Geography=veriler.iloc[:,4:5].values
Gender=veriler.iloc[:,5:6].values
ilkveriler=veriler.iloc[:,3:4].values
sonveriler=veriler.iloc[:,6:13].values
existed=veriler.iloc[:,-1].values

#veri ön işleme
from sklearn.preprocessing import LabelEncoder
le= LabelEncoder()
Geography[:,0]=le.fit_transform(Geography[:,0])
print(Geography)

from sklearn.preprocessing import OneHotEncoder
ohe=OneHotEncoder(categorical_features="all")
Geography=ohe.fit_transform(Geography).toarray()
Gender=ohe.fit_transform(Gender).toarray()
print(Geography)
print(Gender)

```

Kod 1: Veri Ön İşleme

Tablo 1’de veri ön işleme sonrasında elde ettiğimiz, algoritmalar için uygunluğa sahip olan veriyi görmekteyiz.

**Tablo 1:** Veri Ön İşleme sonrası veriye genel bakış

S Custom...	S Surname	I Credits...	I Age	I Tenure	D Balance	I NumOf...	S HasCrC...	S IsActiv...	D Estim...	S Exited	S genderBin	I Spain	I France	I Germany
15634602	Hargrave	619	42	2	0	1	1	1	101,348.88	1	1	0	1	0
15647311	Hill	608	41	1	83,807.86	1	0	1	112,542.58	0	1	1	0	0
15619304	Onio	502	42	8	159,660.8	3	1	0	113,931.57	1	1	0	1	0
15701354	Boni	699	39	1	0	2	0	0	93,826.63	0	1	0	1	0
15737888	Mitchell	850	43	2	125,510.82	1	1	1	79,084.1	0	0	1	0	0
15574012	Chu	645	44	8	113,755.78	2	1	0	149,756.71	1	0	1	0	0
15592531	Bartlett	822	50	7	0	2	1	1	10,062.8	0	1	0	1	0
15656148	Obinna	376	29	4	115,046.74	4	1	0	119,346.88	1	1	0	0	1
15792365	He	501	44	4	142,051.07	2	0	1	74,940.5	0	1	0	1	0
15592389	H?	684	27	2	134,603.88	1	1	1	71,725.73	0	0	0	1	0
15767821	Bearce	528	31	6	102,016.72	2	0	0	80,181.12	0	1	0	1	0

### 3.4 Modelleme ( Modelling )

Modelleme adımı öncelikli olarak Knime Analytics Platform kullanılarak elde edilen işlenmiş veriye sınıflandırma algoritmaları uygulanır. Verinin train ve test olarak bölünmesi esnasında yüzdeler kullanılarak ayrılması yerine X-Partitioner düğümü [11] kullanılmıştır.

#### 3.4.1 Sınıflandırma Algoritmaları

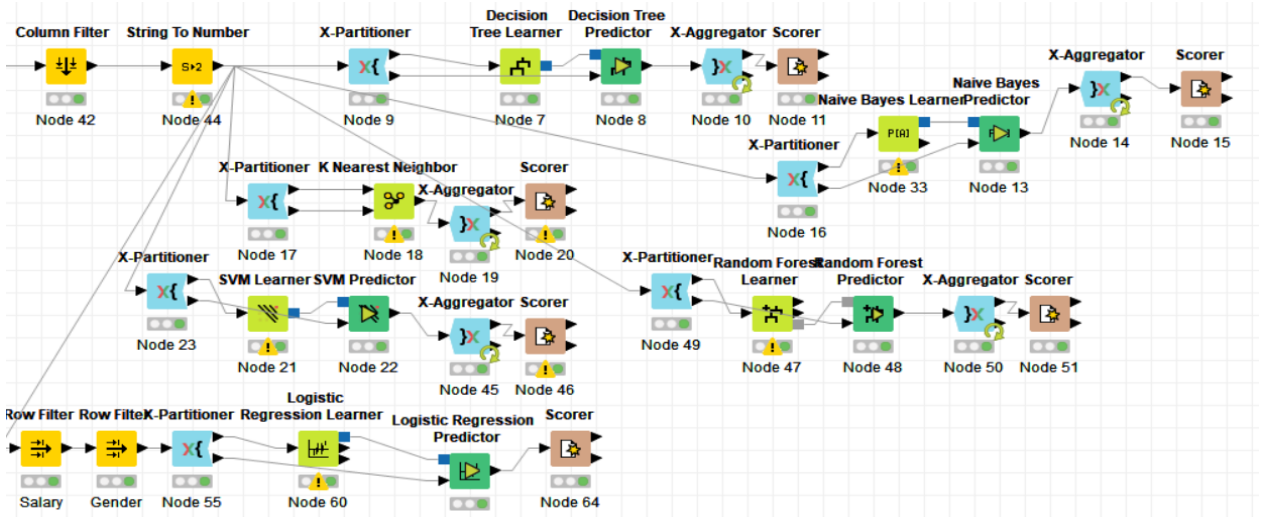
1. Karar ağacı ( Decision Tree ) [12]: İkili veya çok terimli sınıflandırma problemlerini çözmek için klasik bir tahmine dayalı analitik algoritmasıdır.
2. Naif Bayes ( Naive Bayes ) [13]: Düğüm, verilen eğitim verilerinden bir Bayesian modeli oluşturur.
3. K En Yakın Komşu ( K Nearest Neighbor ) [14]: Eğitim verilerini kullanarak K En Yakın Komşu algoritmasına göre bir test verisi kümesini sınıflandırır.
4. SVC (Support Vector Classifier ) [15]: Bu düğüm, verilen değerlerin çıktısını sınıflandırmak için SVC öğrenci düğümü tarafından oluşturulan bir SVC modelini kullanır.
5. Lojistik Regresyon (Logistic Regression) [16]: Çok terimli bir lojistik regresyon gerçekleştirir.
6. Rassal Orman (Random Forest) [17]: Modelleri, rastgele bir orman modelinde tek tek ağaçların tahminlerinin toplanmasına göre tahmin eder.

Uygulanan sınıflandırma algoritmaları ( Decision Tree, Naive Bayes, K Nearest Neighbor, SVC, Logistic Regression, Random Forest ) arasından en yüksek başarı değerini gösteren sınıflandırma algoritması Rassal Orman ( Random Forest ) olur. Karmaşıklık Matrisi ( Confusion Matrix ) kullanılarak oluşturulan sonuçlarda Rassal Orman ( Random Forest ) algoritmasının 86,32% başarı oranı ile en başarılı sınıflandırma algoritması olduğu gözlenmiştir.

**Tablo 2:** Knime Analytics Platform, Algoritmalar

Algoritma	Doğruluk (Accuracy)	Hata Oranı (Error Rate)
Decision Tree	79,88%	20,12%
Naive Bayes	82,39%	17,61%
K-Nearest Neighbor	73,26%	26,73%
SVC	79,63%	20,36%
Logistic Regression	80,9%	19,10%
Random Forest	86,32%	13,68%

Şekil 7’de görüleceği üzere modeller sırasıyla X-Partitioner, sınıflandırma algoritması, X-Aggregator ve Scorer [18] olacak şekilde oluşturulmuştur.

**Şekil 7:** Knime Analytics Platform, Modelleme

Kod 2’de Knime modelleri arasından en başarılı olan Rassal Orman (Random Forest), Spyder geliştirme ortamında Python dili ile modellenmiştir. Kriter (Criterion) ‘gini’ ve ‘entropy’ olarak 2 şekilde denenmiş, ‘entropy’ kriteri daha çok doğru tahminde bulunarak daha başarılı olmuştur.

```
from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier(n_estimators=10, criterion = 'entropy')
rfc.fit(X_train,y_train)
```

**Kod 2:** Rassel Orman (Random Forest)

Kod 3’de K Katlamalı Çapraz Doğrulama ( K Fold Cross Validation) [19] kullanılarak modelin iyileştirilmesini görmektesiniz. K Katlamalı Çapraz Doğrulama ( K Fold Cross Validation) ortalama başarısı 0.8489, standart sapması 0.0024 olarak gözlemlenmiştir.

```
#k-fold cross validation:
from sklearn.model_selection import cross_val_score
basari=cross_val_score(estimator=rfc,X=X_train,y=y_train,cv=4)
print(basari.mean()) #0.8489557046845906
print(basari.std()) #0.0024625168985655784
```

**Kod 3:** K Katlamalı Çapraz Doğrulama ( K Fold Cross Validation)

### 3.5 Değerlendirme ( Evaluation )

Modelin pozitif ve negatif örnekleri içerisinde barındıran ve test veri setini ne ölçüde sınıflandırdığını gösteren Karmaşıklık Matrisi ( Confusion Matrix ) [20] sonuçları algoritmanın değerlendirilmesi açısından önemlidir.

**Tablo 3:** Karmaşıklık Matrisi ( Confusion Matrix )

	Öngörülen Sınıf Negatif:	Öngörülen sınıf Pozitif:
Gerçek Sınıf Negatif:	TN:2506	FP:111
Gerçek Sınıf Pozitif:	FN: 380	TP:303

True Negative (TN): Gerçek Negative

False Negative (FN): Yanlış Negatif

False Positive (FP): Yanlış Pozitif

True Positive (TP): Gerçek Pozitif

$$\text{Modelin Duyarlılığı ( Sensivity-Precision )} = \frac{TP}{TP+FN} = 0.44$$

$$\text{Modelin Belirleyiciliği ( Specificity )} = \frac{TN}{TN+FP} = 0.95$$

$$\text{Modelin Kesinliği ( Recall )} = \frac{TP}{TP+FP} = 0.73$$

Modelin Yanlış Sınıflandırma Oranı (Misclassification Rate) =  $\frac{FP+FN}{TOPLAM} = 0.14$  Aynı zamanda hata oranı (Error rate) olarak adlandırılır.

F ölçütü kesinlik ve duyarlık performans değerlendirme ölçütlerinin harmonik ortalamasıdır. Her iki ölçütü birlikte değerlendirme imkanı verir.

$$F \text{ ölçütü} = \frac{2*Precision*Recall}{Precision+Recall} = 0.549$$

$$\text{Modelin Doğruluğu (Accuracy)} = \frac{TP+TN}{TOPLAM} = 0.85$$



Cohen'in kappa katsayısı (Cohen's Kappa) iki değerleyici arasındaki karşılaştırmalı uyuşmanın güvenilirliğini ölçen bir istatistik yöntemidir [21].

Modelin Cohen'in kappa katsayısı (Cohen's Kappa) = 0.5

Tablo 3'de modelin 0, 1 tahmini ve makro ortalama (etiket başına ortalama ağırlık ortalaması), ağırlık ortalama (etiket başına ortalama ağırlık ortalaması), micro ortalama (toplam gerçek pozitiflerin, hatalı negatiflerin ve yanlış pozitiflerin ortalaması) duyarlılık, kesinlik, F1 ve destek skoru gösterilmiştir. İkili sınıflandırmada, pozitif sınıfın (1) keskinliği (Recall) “duyarlılık (precision)” olarak da bilinir; Negatif sınıfın (0) keskinliği (Recall) ise “belirleyicilik (specifity)” olarak adlandırılır. Sınıflandırma raporu hesaplamaları tek bir tabloda çıkartıp, yorumlamayı kolaylaştırdığı için kullanımı avantajlıdır.

**Tablo 4:** Sınıflandırma Raporu ( Classification Report )

Report	precision	recall	f1-score	support
0	0.87	0.96	0.91	2617
1	0.73	0.46	0.56	683
micro avg	0.85	0.85	0.85	3300
macro avg	0.80	0.71	0.74	3300
weighted avg	0.84	0.85	0.84	3300

### 3.5.1 OptiScorer Karşılaştırması

OptiWisdom çatısı altındaki üç ana motordan biri olan Optiscorer Engine, yapay zekaya dayanarak birçok alanda skorlama yapabilir [22]. OptiScorer Engine'den alınan skorlar Excel Solver kullanılarak optimize edilmiş eşik değeri kullanılarak sınıflandırılır. Optimize edilmiş eşik değeri 0,34 olarak hesaplanmıştır. Eşik değerinin üzerindeki değerler 1, altındakiler 0 olmak üzere sınıflandırma gerçekleştirilir. Sınıflandırma sonucunda Karmaşıklık Matrisi ( Confusion Matrix ) çıkarılır ve Tablo 4'de görülmekte olan hesaplamalar yapılır. Banka müşteri verisi üzerinde uygulanmış olan Rassal Orman ( Random Forest ) algoritması ile aynı veri üzerinde OptiScorer Engine uygulaması sonuçları karşılaştırılır. OptiScorer Engine 90% doğruluk yüzdesi ve 0,757 F1 Skoru ile sınıflandırmada daha başarılı olmuştur.

**Tablo 5:** Rassal Orman Algoritması ile OptiScorer Engine Karşılaştırması

	OptiScorer	Rassal Orman (Random Forest)
Doğruluk (Accuracy)	0,90	0,85
Yanlış Sınıflandırma Oranı (Error Rate)	0,10	0,14
Duyarlılık (Sensitivity)	0,76	0,44
Belirleyicilik (Specifity)	0,94	0,95
Keskinlik (Recall)	0,76	0,73
F Skoru (F Score)	0,75	0,54

### **3.6 Canlıya Alma (Deployment)**

CRISP-DM metodolojisinin son adımı olan uygulama adımında ayrılma potansiyeli keşfedilen banka müşterilerinin bilgileri müşteri ilişkileri yönetimi ( Customer Relationship Manegement ) kapsamında değerlendirilir ve kişilere özel sadakat programları geliştirilir.

### **4.Sonuç**

Yapılan çalışmada veri madenciliği teknikleri kullanılarak bankadan ayrılarak rakip bankanın müşterisi olmayı tercih eden müşteri davranışlarını ortaya koyan modeller geliştirilmiştir. Bu modeller oluşturulurken verinin yapısı göz önüne alınarak sınıflandırma algoritmaları kullanılmıştır. Knime Analytics Platfrom kullanılarak verinin temizlenmesi gerçekleştirilmiş ardından yine aynı platform kullanılarak sınıflandırma algoritmaları ( Decision Tree, Naive Bayes, K Nearest Neighbor, SVC, Logistic Regression, Random Forest ) uygulanmıştır. Bu algoritmaların doğruluk ( Accuracy ) dereceleri karşılaştırılarak sınıflandırma için en uygun algoritmanın Rassal Orman ( Random Forest ) olduğu saptanmıştır. Spyder geliştirme ortamında Python dili ve Knime Analytics Platform kullanılarak Rassal Orman (Random Forest) ile müşterinin ayrılarak rakip şirketi tercih etme ( Churn ) davranışı modellenmiş %85 doğruluk oranıyla başarıya ulaşılmıştır. Optimum sonuç elde edilebilmesi için müşteri verisi OptiWisdom ürünlerinden biri olan OptiScorer ile ayrıca skorlanmıştır. OptiScorer yapay zekaya dayanan skorlama sistemiyle doğruluk yüzdesini 90% 'a kadar çıkartmıştır.

Bu elde edilen sonuçlar ışığında bankanın ayrılma potansiyeline sahip müşterilerin müşteri ilişkileri yönetimi ( Customer Relationship Manegement ) kapsamında değerlendirmesi ve sadakat programlarının uygulanması tavsiye edilir. Bu programların uygulanması halinde banka müşterilerini elde tutabilecektir. Makale; bankacılık dışında müşteri kaybının önemli olduğu tüm sektörlerde de uygulanacak tahmin yöntemlerinin bir adım daha ileriye götürülmesi ile gelecek çalışmalara katkı sağlayabilecektir.

## Kaynakça

- [1] Massey P., Mitzi M., “ Reengineering CRM: Leveraging Knowledge Assets In IBM”, Decision Support Systems, Cilt 32, No 2, 155-170, 2001.
- [2] Vildan GÜLPINAR, Yapay Sinir Ağları ve Sosyal Ağ Analizi Yardımı ile Türk Telekomünikasyon Piyasasında Müşteri Kaybı Analizi, Marmara Üniversitesi İ.İ.B. Dergisi YIL 2013, CİLT XXXIV, SAYI I, S. 331 - 350
- [3] Hudairy H., “Data Mining and Decision Making Support In The Governmental Sector”, Master Thesis, Louisville University, 2004. Surname A and Surname B 2009 Journal Name 23 544
- [4] Şadi Evren ŞEKER, İş Zekası (Business Intelligence), YBS Ansiklopedi, v. 3, is. 1, March, 2016, pp. 20 - 25
- [5] Şadi Evren ŞEKER, İş Zekası ve Veri Madenciliği, İstanbul, Cinius Yayınları, ISBN 978-605-127-671-7, 2013
- [6] <https://pazarlamasyon.com/yeni-nesil-sadakat-programlari-nasil-dizayn-edilir-ve-basarili-olur/>
- [7] Emel KIZILKAYA AYDOĞAN, CHURN ANALYSIS AND CUSTOMER SEGMENTATION OF A COSMETICS BRAND USING DATA MINING TECHNIQUES, Sigma Vol./Cilt 26 Issue/Sayı 1
- [8] Michael R. Berthold, Nicolas Cebron, Fabian Dill, Thomas R. Gabriel, Tobias Kotter, Thorsten Meinl, Peter Ohl, Kilian Thiel and Bernd Wiswedel, KNIME - The Konstanz Information Miner
- [9] <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html>
- [10] <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OneHotEncoder.html>
- [11] <https://hub.knime.com/knime/extensions/org.knime.features.base/latest/org.knime.base.node.meta.xvalidation.XValidatePartitionerFactory>
- [12] <https://www.knime.com/knime-introductory-course/chapter6/section3/decision-tree>
- [13] <https://hub.knime.com/knime/extensions/org.knime.features.base/latest/org.knime.base.node.mine.bayes.naivebayes.learner3.NaiveBayesLearnerNodeFactory4>
- [14] <https://hub.knime.com/knime/extensions/org.knime.features.base/latest/org.knime.base.node.mine.knn.KnnNodeFactory2>
- [15] <https://hub.knime.com/knime/extensions/org.knime.features.base/latest/org.knime.base.node.mine.svm.predictor2.SVMPredictorNodeFactory>
- [16] <https://hub.knime.com/knime/extensions/org.knime.features.base/latest/org.knime.base.node.mine.regression.logistic.learner4.LogRegLearnerNodeFactory4>
- [17] <https://hub.knime.com/knime/extensions/org.knime.features.base/latest/org.knime.base.node.mine.treeensemble2.node.randomforest.predictor.classification.RandomForestClassificationPredictorNodeFactory2>
- [18] <https://hub.knime.com/knime/extensions/org.knime.features.base/latest/org.knime.base.node.mine.scorer.accuracy.AccuracyScorerNodeFactory>
- [19] [https://scikit-learn.org/stable/modules/generated/sklearn.model\\_selection.KFold.html](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.KFold.html)
- [20] [https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion\\_matrix.html](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html)
- [21] Matthijs J Warrens, Five Ways to Look at Cohen’s Kappa, Warrens, J Psychol Psychother 2015, 5:4
- [22] <https://www.optiwisdom.com/>