

IEEE-CIS Fraud Detection

Güldane DURUKAN
Selim BEYHAN
Talip EROĞLU

İş Problemi

- Kredi kartı aracılığıyla online yapılan işleme ilişkin çeşitli bilgiler verildiğinde bu işlemin dolandırıcılık olma olasılığını hesaplayan bir makine öğrenmesi modeli kurmak
- Bu model sayesinde Dolandırıcılık işlemleri doğru tespit edilerek işlem gerçekleşmeden kartın bloklaması amaçlanmakta,
- Dolandırıcılık işlemleri “dolandırıcılık değil” şeklinde tahmin edilmemeli, aynı şekilde dolandırıcılık olmayan işlemler de dolandırıcılık olarak etiketlenip kart gereksiz yere bloklanmamalı

Yol Haritası

- Öncelikle Keşifçi Veri Analizi yaparak değişkenlerin etkisinin analizi ve verinin sadeleştirilmesi
- Bir temel model oluşturulması
- Yeni özellikler oluşturularak bu özelliklerin modelin performansını iyileştirdiğinin analizinin yapılması

Değişkenler

Transaction Veri Seti

- **Transaction id:** İşleme ilişkin unique id bilgisi
- **TransactionDT:** belirli bir referans zamana olan uzaklık
- **TransactionAMT:** USD cinsinden ödeme tutarı
- **ProductCD [Categorical]:** ürün kodu
- **card1–6 [Categorical]:** kart türü, ülke vb. gibi ödeme kartıyla ilgili bilgiler
- **addr1, addr2 [Categorical]:** adres bilgisi
- **dist1, dist2:** uzaklık bilgisi
- **P_emaildomain [Categorical]:** müşterinin email domain i
- **R_emaildomain [Categorical]:** parayı alanın email domain i

- **C1-C14:** ödeme kartı ile kaç adresin ilişkilendirildiği vb. sayı bilgisi
- **D1-D15:** time delta, önceki işlemler arasında geçen zaman bilgisi (gün vb.)
- **M1-M9 [Categorical]:** eşleşme bilgisi, karttaki isimler ve adres vb.
- **Vxxx:** Vesta tarafından geliştirilmiş çeşitli özellikler

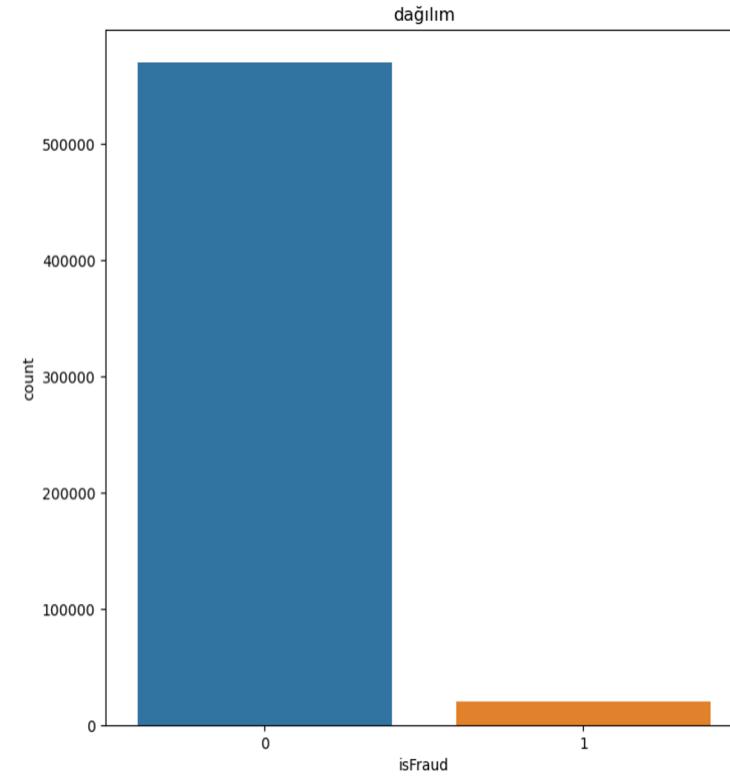
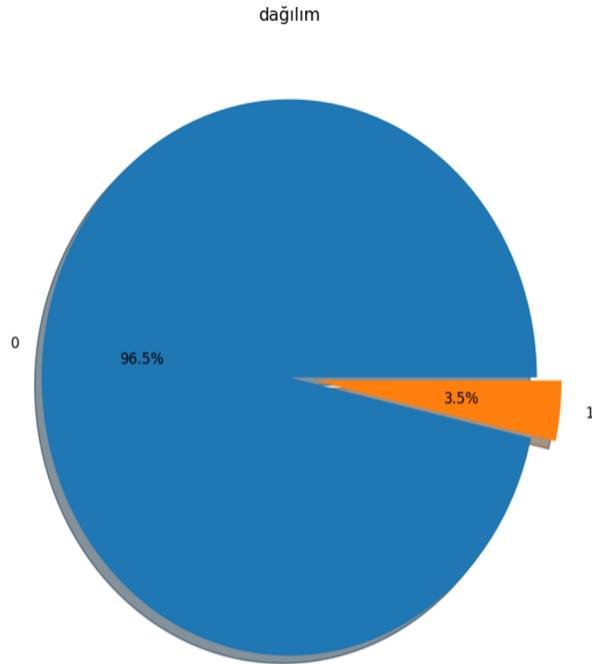
Identity Veri Seti

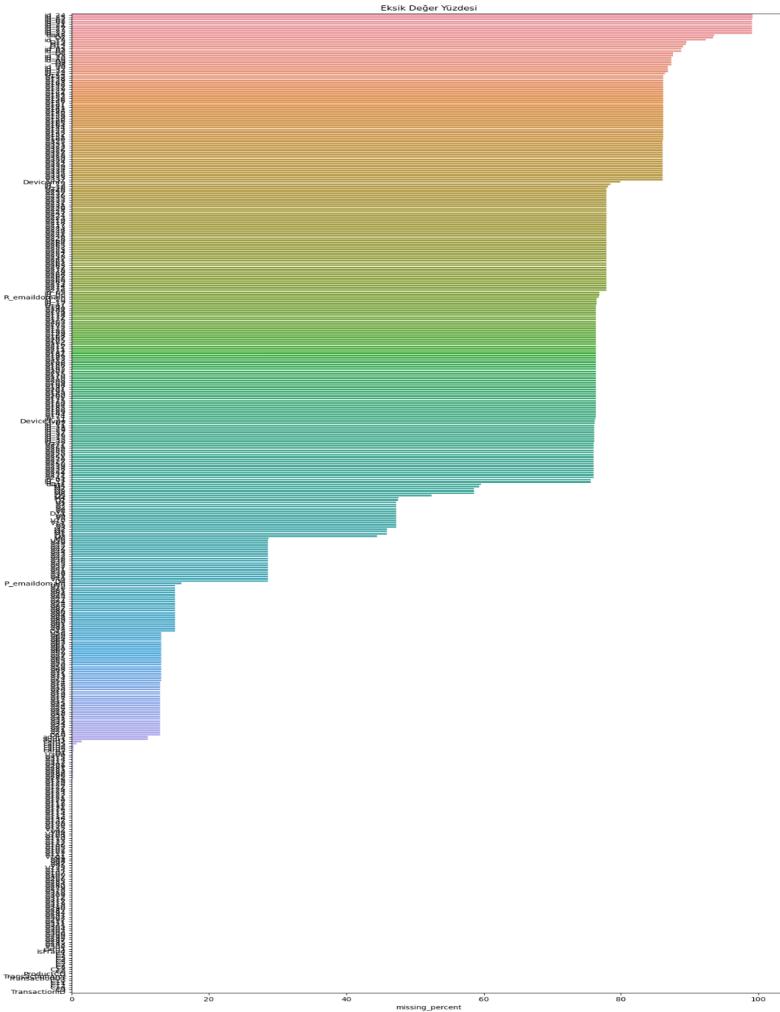
- **Transaction id:** İşleme ilişkin unique id bilgisi
- **DeviceType [Categorical]:** İşlem için kullanılan cihaz tipi
- **DeviceInfo [Categorical]:** Kullanılan cihaz hakkında daha fazla bilgi
- **id 1–38 [Categorical+numeric]:** ağ bağlantısı bilgisi, tarayıcı bilgisi vb kategorik bilgiler

Kesifçi Veri Analizi

	<i>Train</i>	<i>Test</i>
•Gözlem Sayısı:	<i>590540</i>	<i>506691</i>
•Değişken Sayısı:	<i>434</i>	<i>433</i>
•Kategorik Değişken:	<i>112</i>	<i>111</i>
•Numerik Değişken:	<i>316</i>	<i>326</i>
•Kategorik fakat Kardinal Değişken:	<i>6</i>	<i>6</i>
•Numerik fakat Kategorik:	<i>87</i>	<i>76</i>

Hedef DEĞİŞKEN (is Fraud)



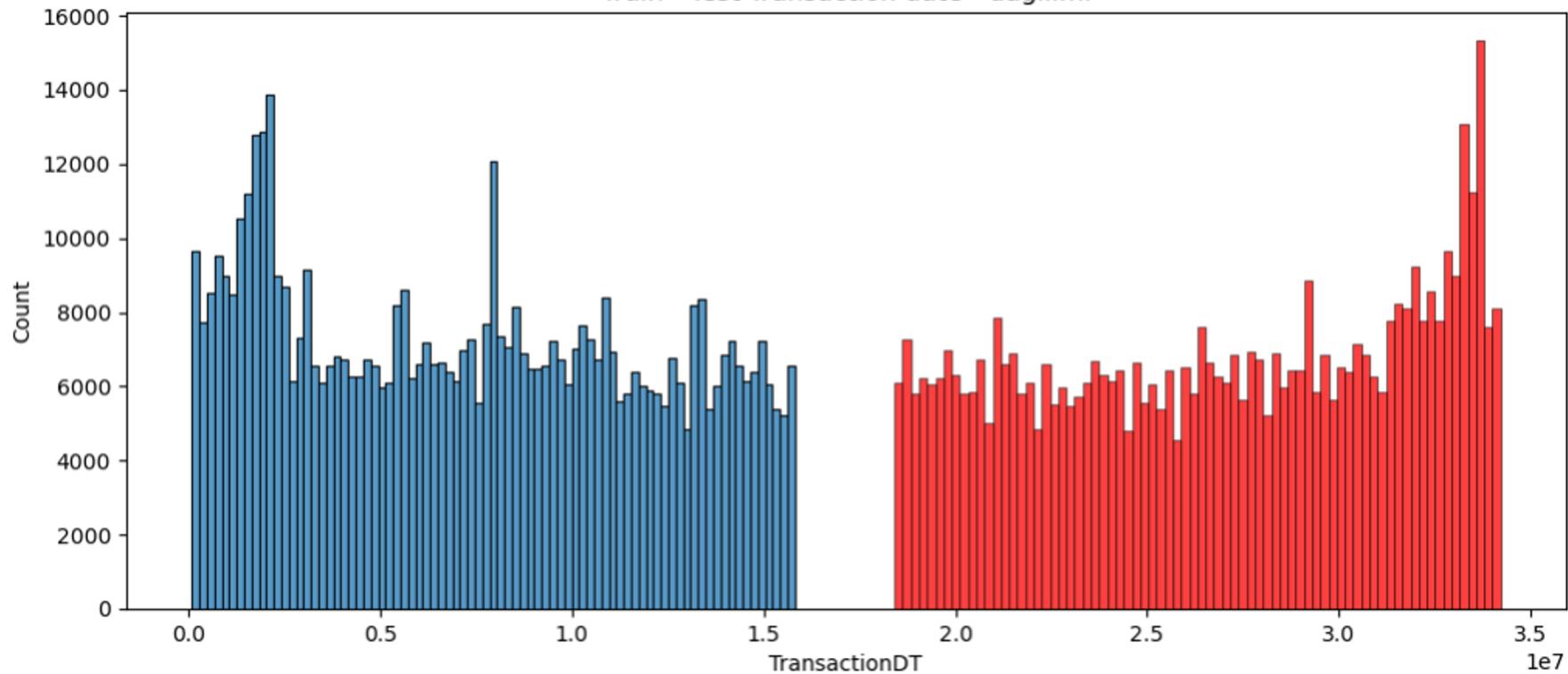


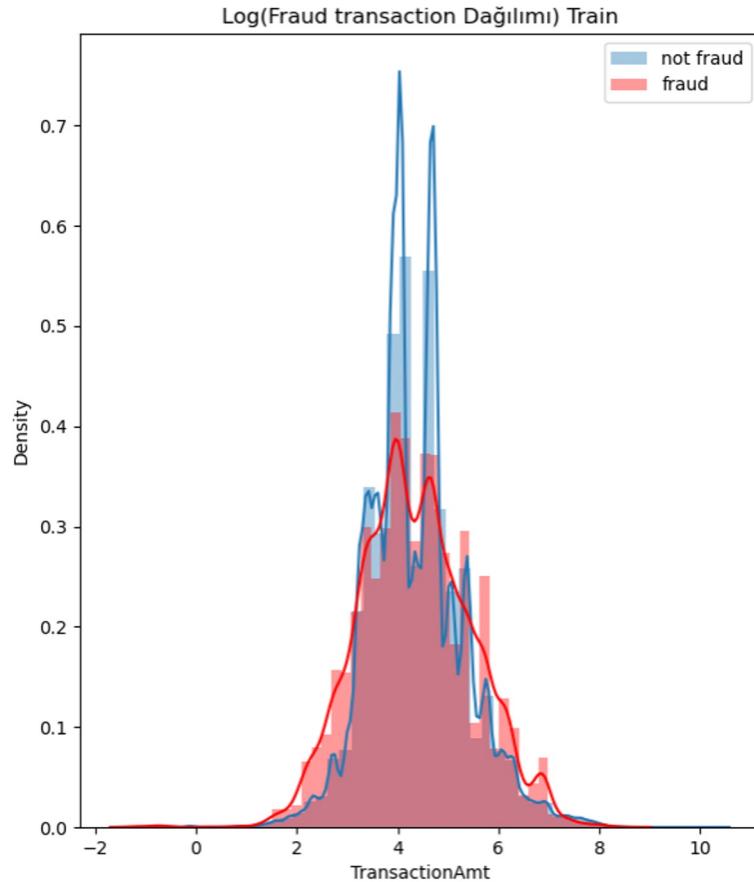
```
missing_values_table(df_train)
```

	n_miss	ratio
id_24	585793	99.200
id_25	585408	99.130
id_07	585385	99.130
id_08	585385	99.130
id_21	585381	99.130
...	...	
V285	12	0.000
V284	12	0.000
V280	12	0.000
V279	12	0.000
V312	12	0.000

- 414 değişken eksik değer içeriyor
- 214 değişkenin %50inden fazlası NA

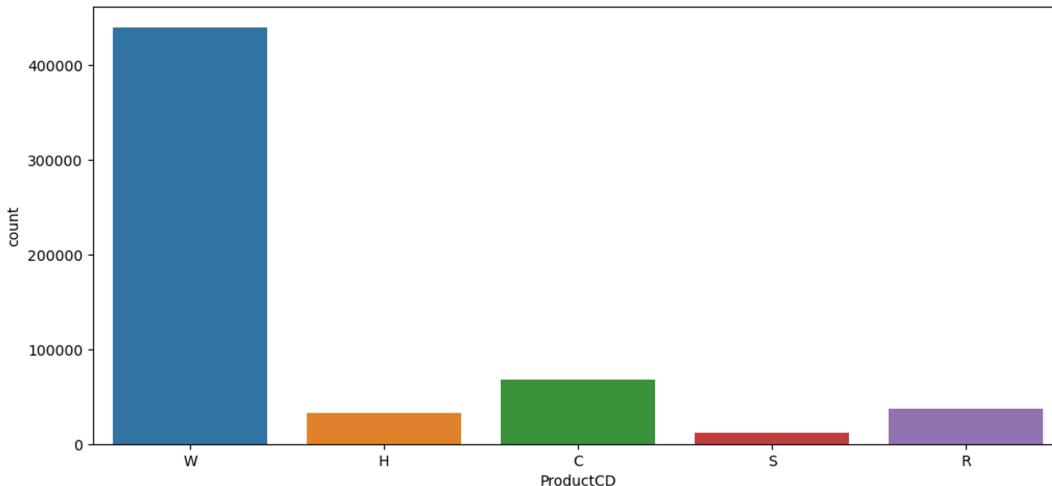
Train - Test Transaction date - dağılımı



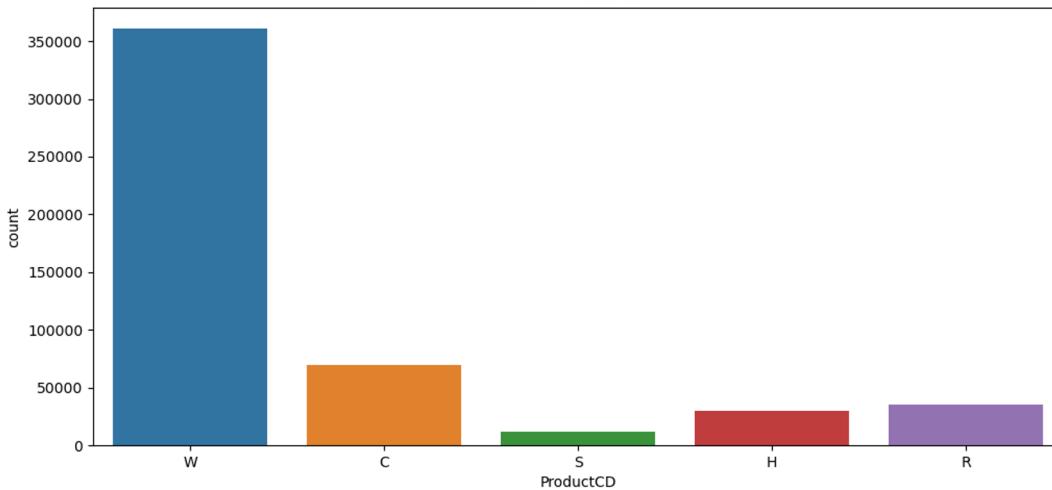


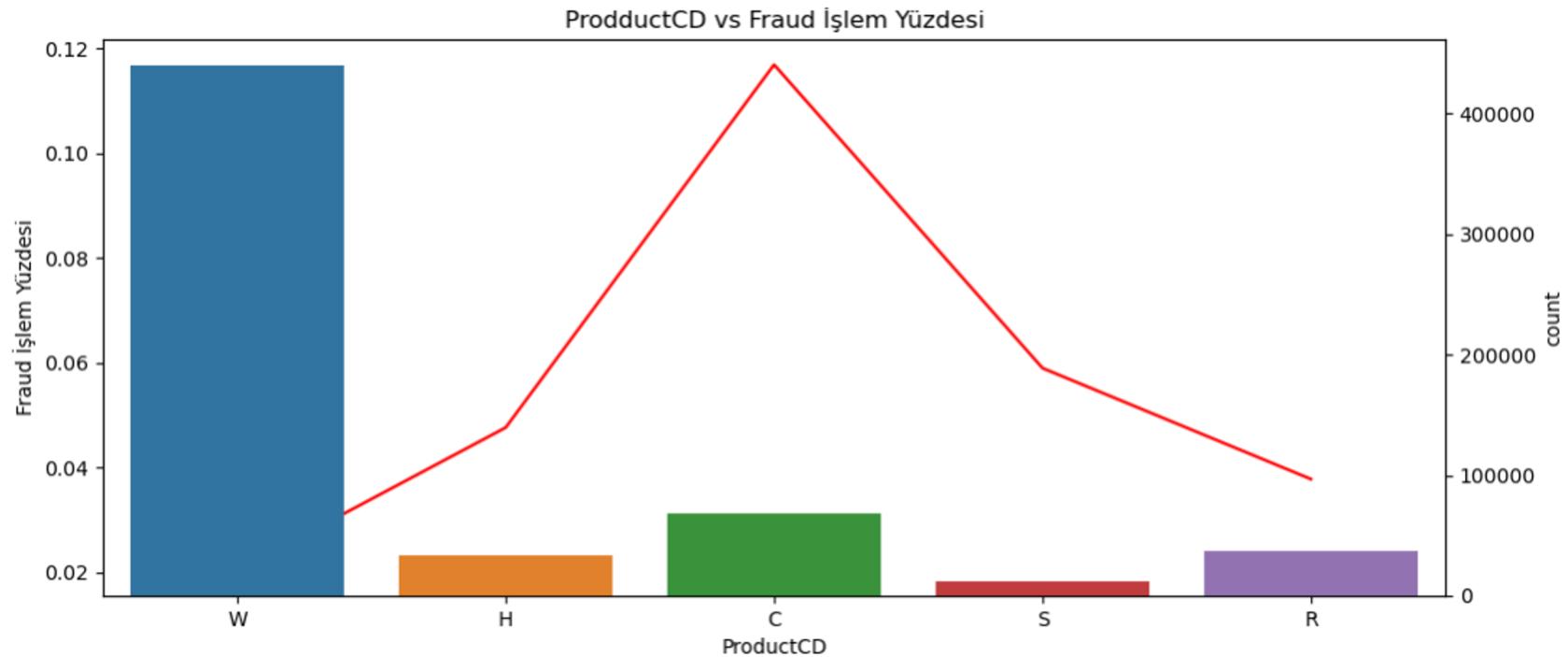
- Ödeme Tutarı 27\$($\log=3.3$) dan küçük 244\$($\log=5.5$) dan büyük olan işlemlerin fraud olma olasılığı daha yüksek.
- Diğer taraftan 27\$ ile 244\$ arasındaki işlemlerde ise “yasal” olma olasılığı daha yüksek

Train ProductCD



Test ProductCD

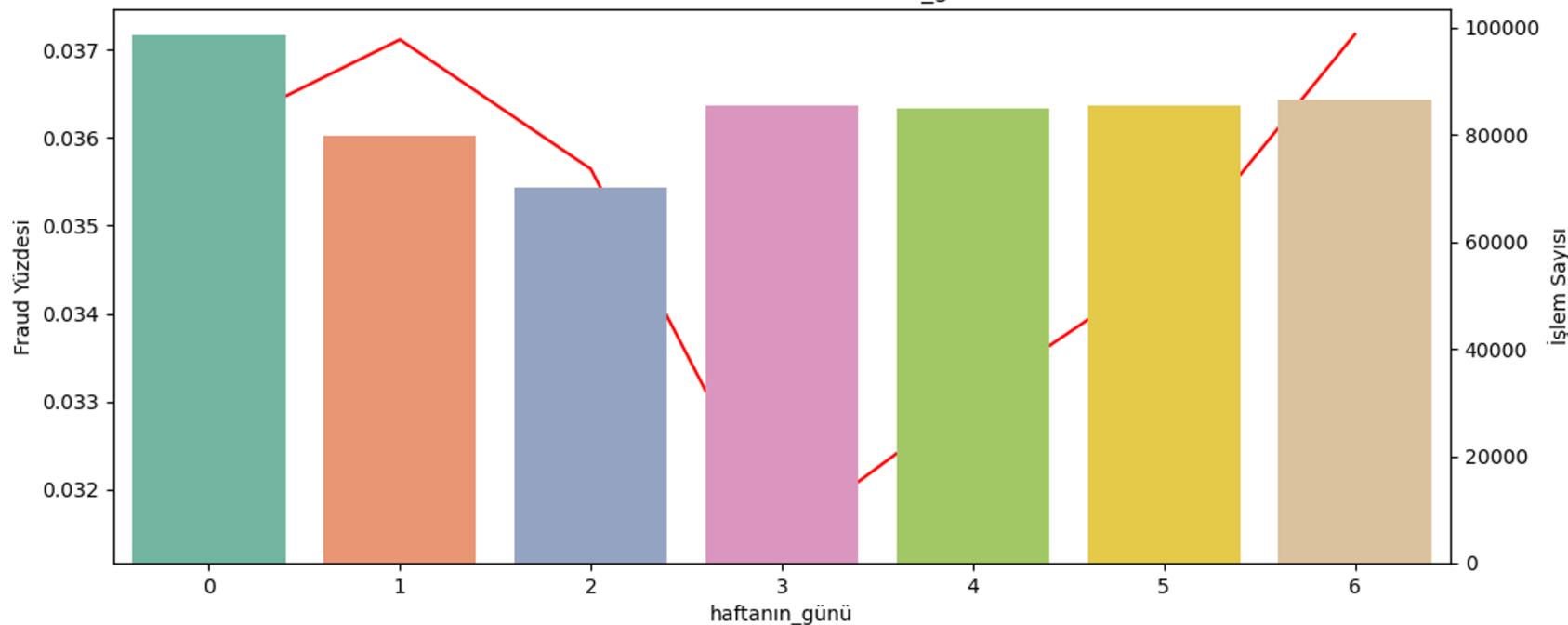




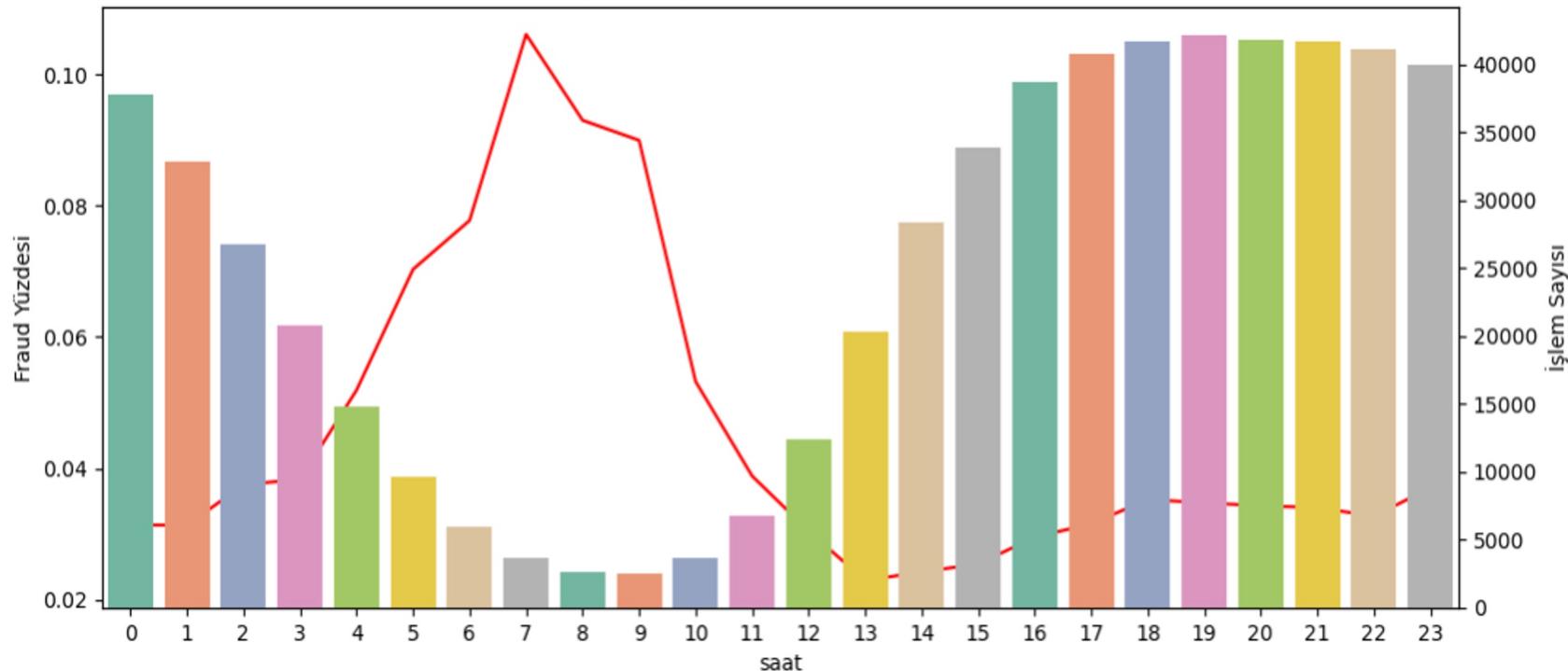
Tüm işlemler arasında W kategorisi en fazla yer almaktadır.

- C kategorisinde yaklaşık 12% Fraud oranı
- S kategorisinde yaklaşık 6% Fraud oranı
- H kategorisinde yaklaşık 5% Fraud oranı
- W kategorisinde yaklaşık 2% Fraud oranı

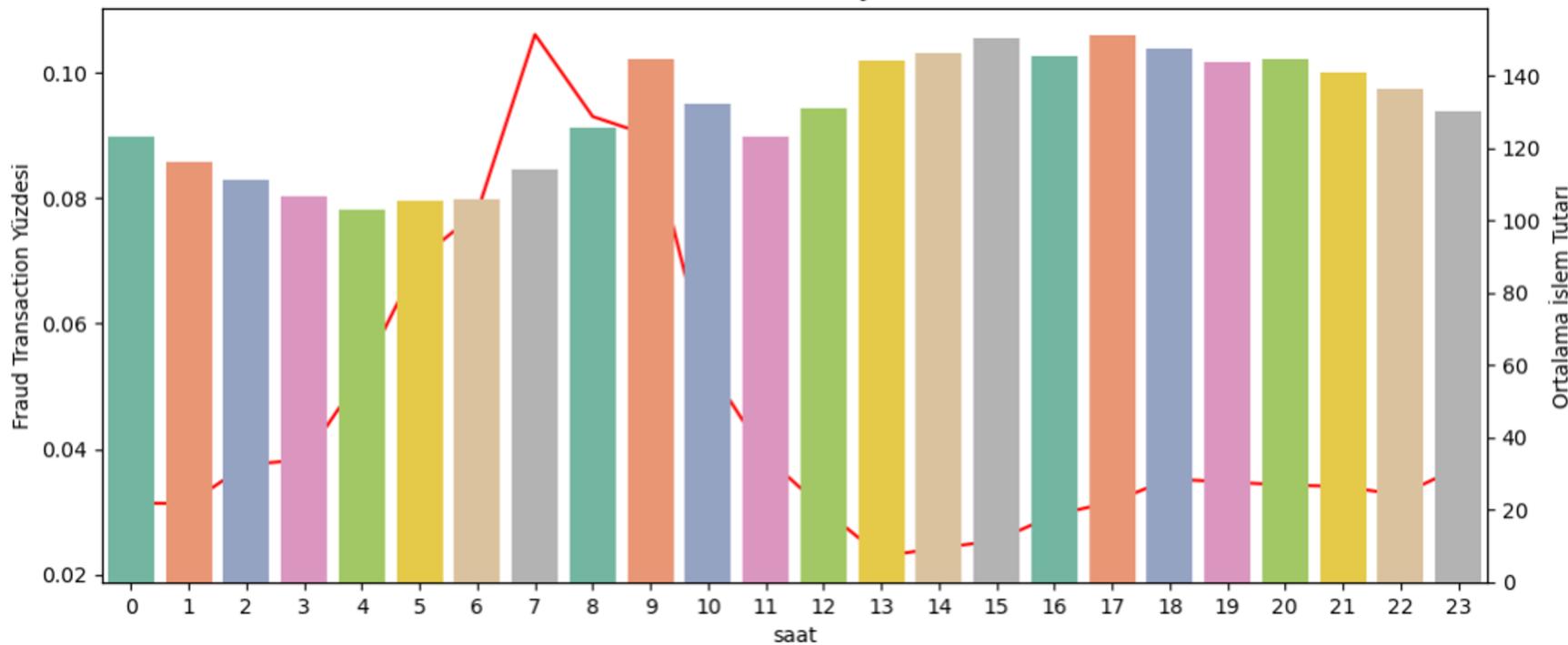
Fraud transaction vs haftanın_günü



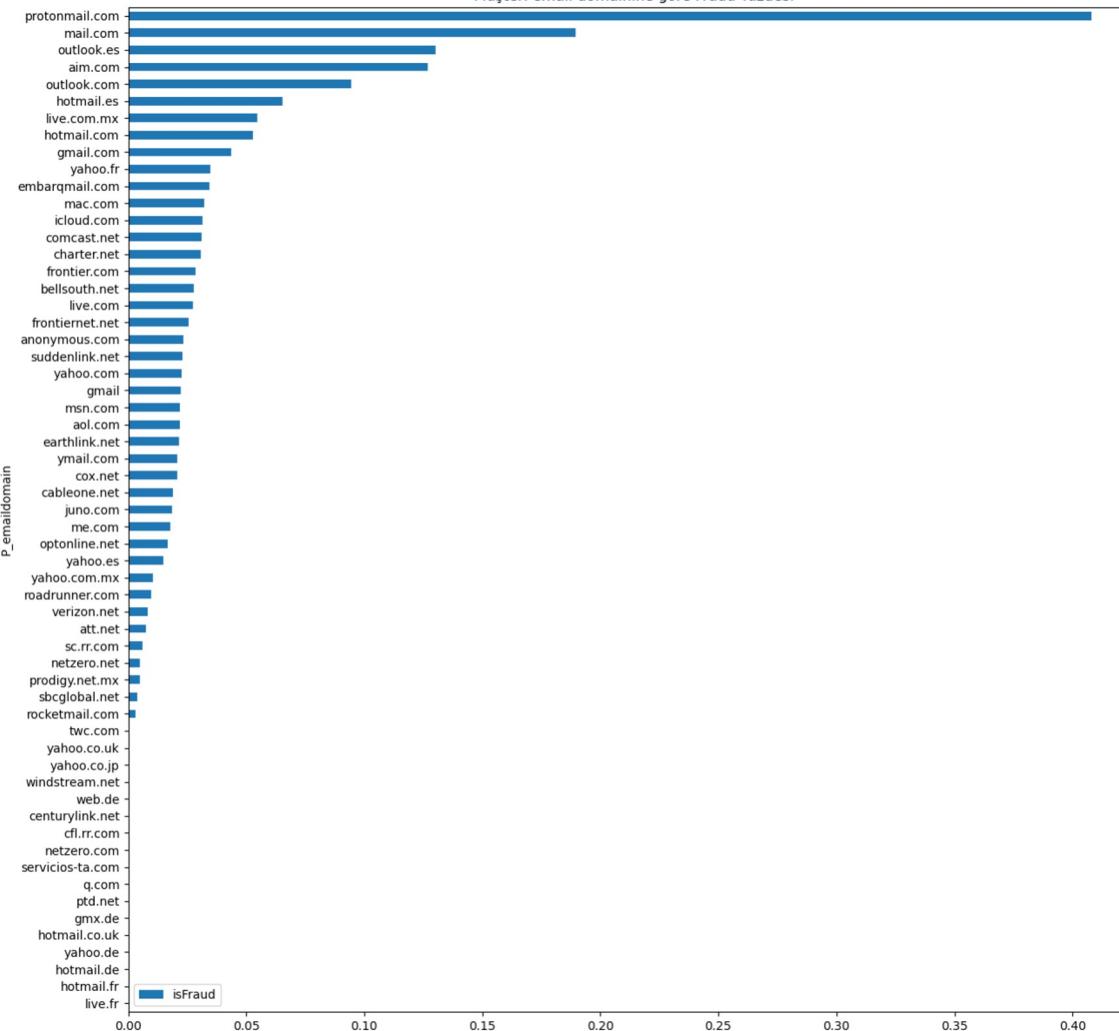
İşlem Sayısı-Fraud Yüzdesi vs saat



Fraud transaction-Ortalama işlem Tutarı vs saat

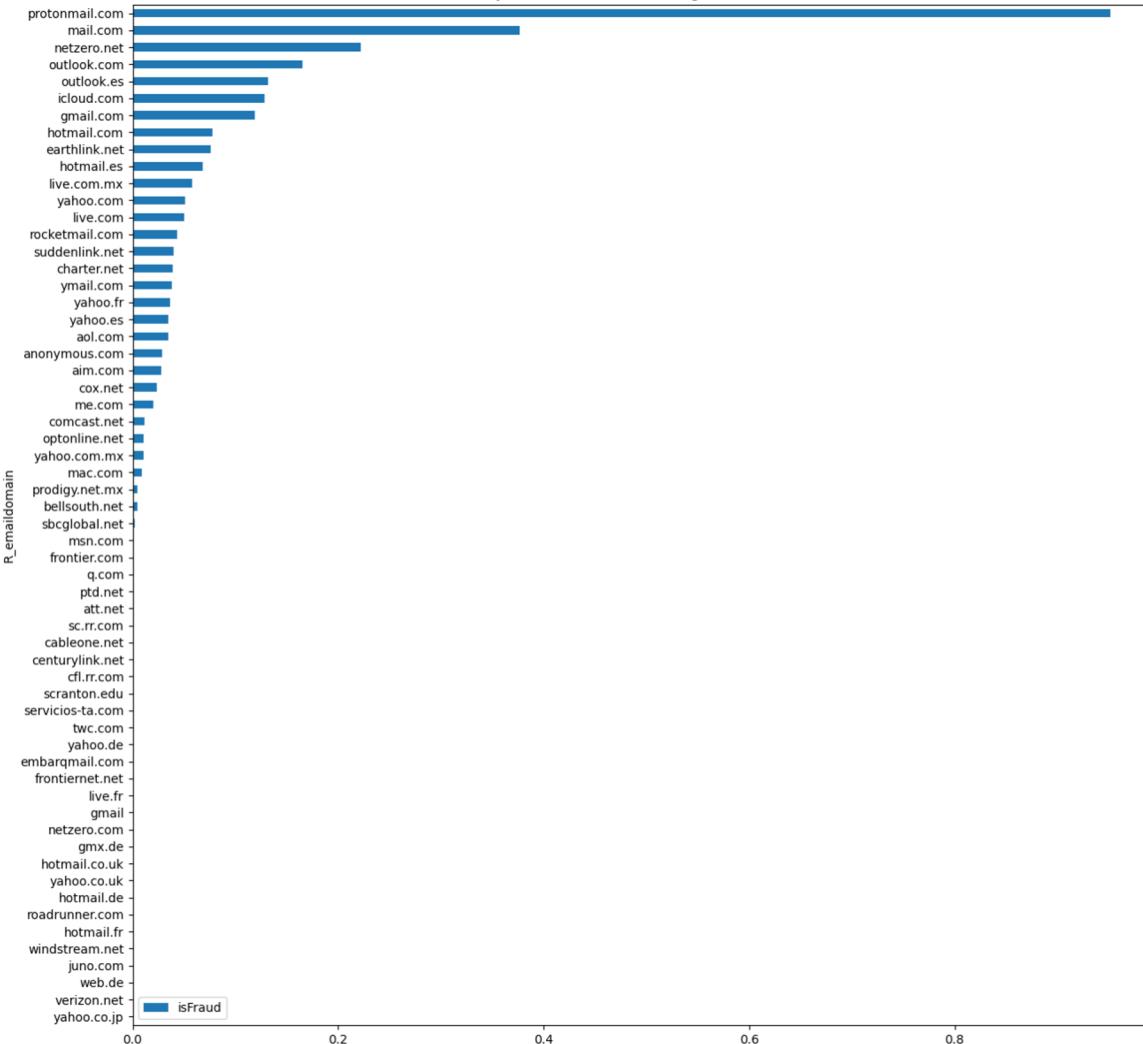


Müşteri email domainine göre Fraud Yüzdesi



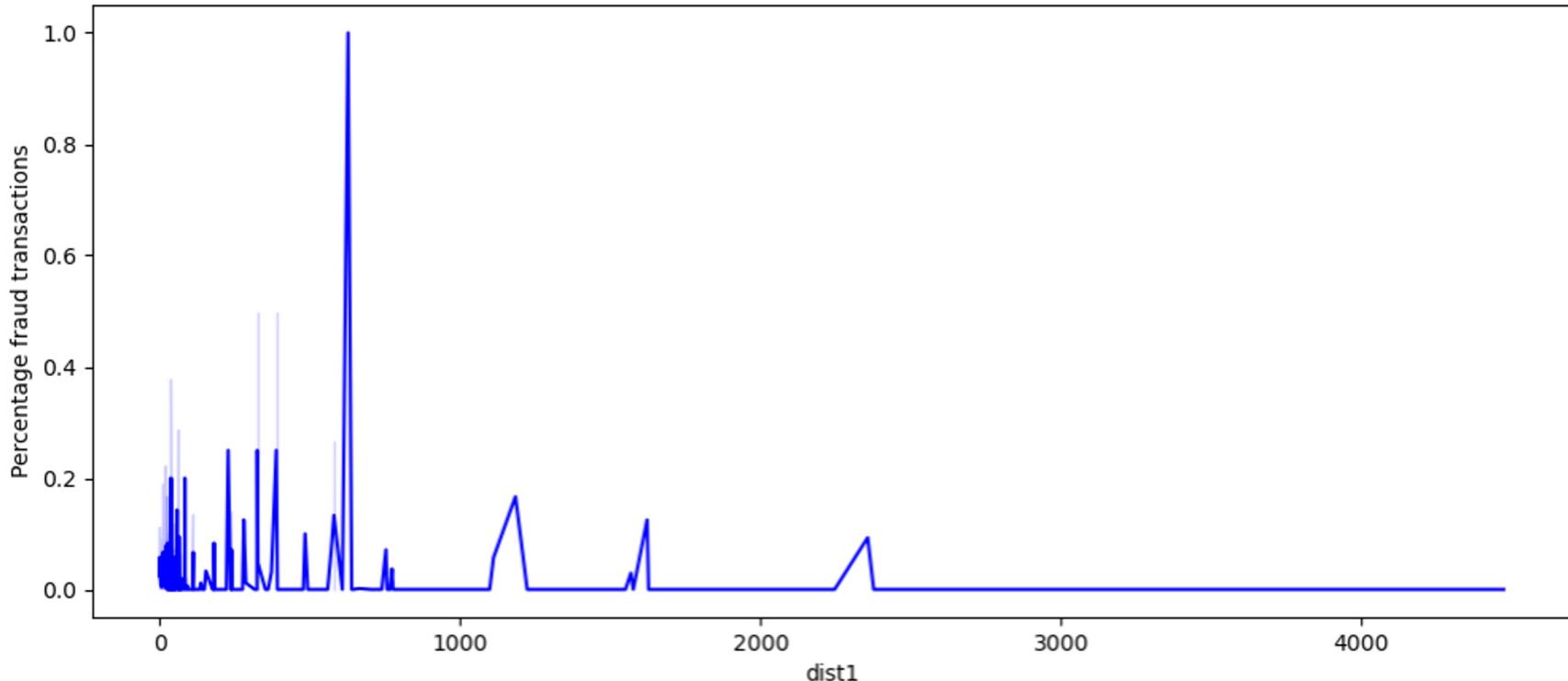
Müşteri email domainine göre
Fraud Yüzdesi

Parayı alanın email domainine göre Fraud Yüzdesi

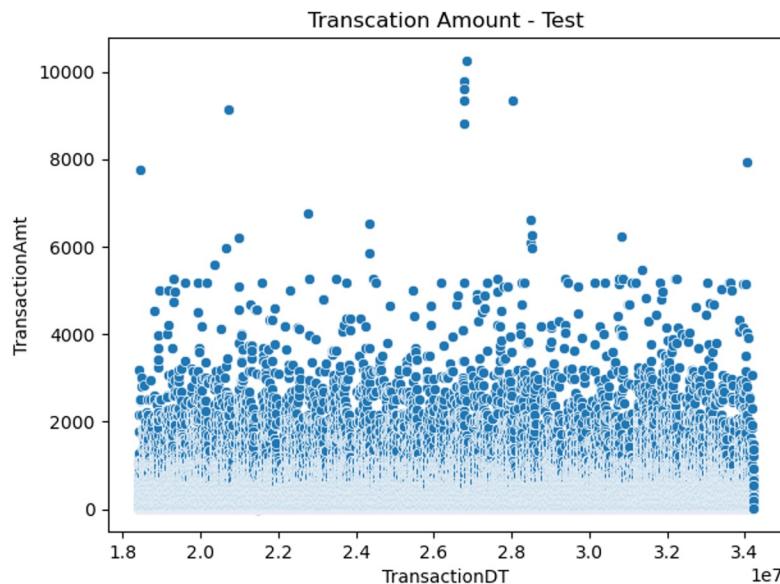
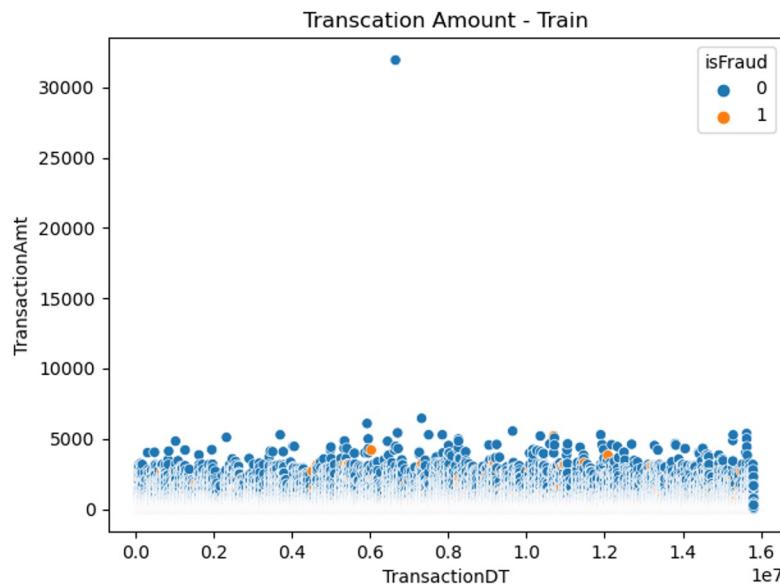


Parayı alanın email domainine göre
Fraud Yüzdesi

Dist 1 e göre Fraud Transaction yüzdesi



Aykırı Değer Analizi



Özellik Mühendisliği

Eksik Değerler

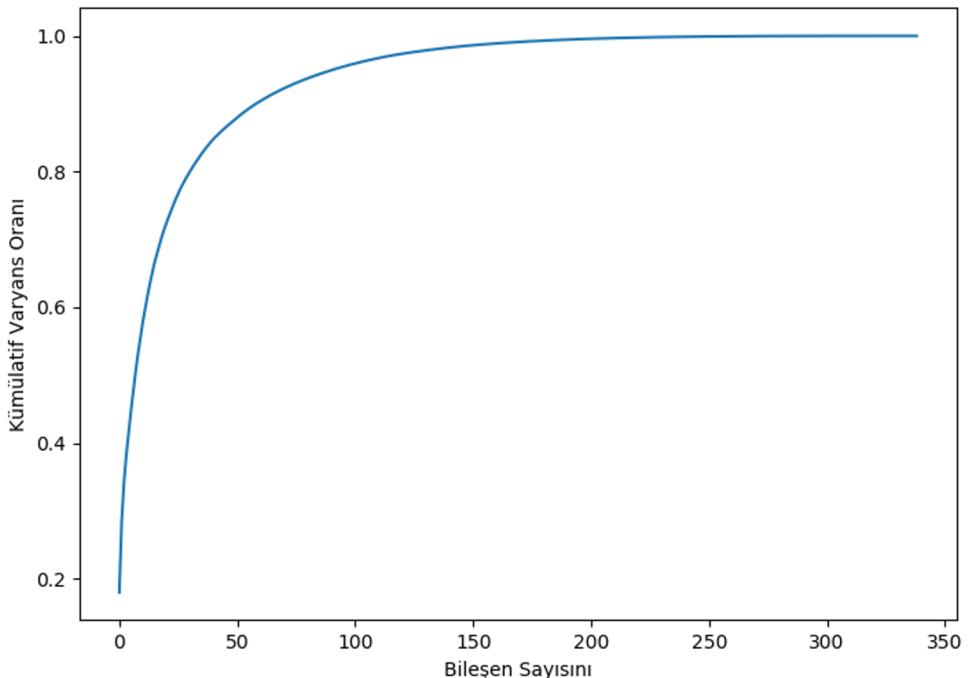
Yüzde 90 üzerinde eksik değer barındıran değişkenler veri setinden çıkarıldı

Eksik değerler -999 ile dolduruldu.(Ağaç yöntemlerinde modelin ilgili gözlemin gerçek bir değer olmadığını anlayarak bölünme sırasında o değere farklı davranışması için)

Aykırı Değerler

%5 ve % 95 lik quartile lar dikkate alınarak üç değerlere baskılandı

V_columns-PCA



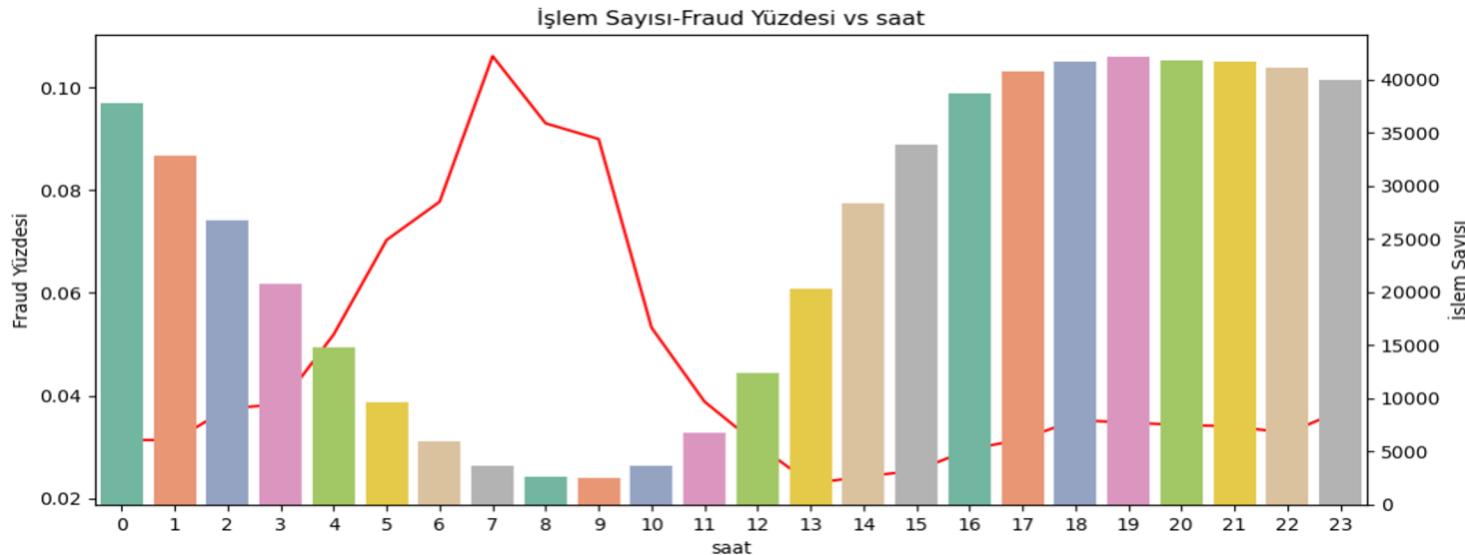
```
array([0.13151555, 0.24568397, 0.30691201, 0.35565875, 0.39881565,
       0.43425891, 0.46508895, 0.49444105, 0.52152347, 0.54728742,
       0.56990279, 0.59144289, 0.61085557, 0.62836593, 0.6449951 ,
       0.66076592, 0.67545715, 0.68940085, 0.70297572, 0.71514765,
       0.726812 , 0.73746012, 0.747136 , 0.75669117, 0.76560158,
       0.7740171 , 0.78165455, 0.78921679, 0.7961483 , 0.80288521,
       0.80914441, 0.81530517, 0.82141704, 0.82678334, 0.83184027,
       0.83680119, 0.84142188, 0.84584956, 0.85001907, 0.85415359,
       0.8580631 , 0.86188568, 0.86555037, 0.86921074, 0.87272191,
       0.87610801, 0.87944865, 0.88267845, 0.88583342, 0.88893627,
       0.89190491, 0.89475769, 0.89751357, 0.90019921, 0.90286512,
       0.90540782, 0.90791609, 0.91030137, 0.91264516, 0.91489153,
       0.91704948, 0.91915785, 0.92119746, 0.92317486, 0.92511395,
       0.92700857, 0.92882838, 0.93063099, 0.93238818, 0.93411498,
       0.93580378, 0.93740436, 0.93899947, 0.9405467 , 0.94205226,
       0.94352701, 0.94494787, 0.94635814, 0.94772974, 0.94907233,
       0.95036263, 0.95159952, 0.95282199, 0.9539812 , 0.95511315,
       0.95621415, 0.95729398, 0.9583568 , 0.95936455, 0.96035953,
       0.96134037, 0.96227144, 0.96319173, 0.96410097, 0.96499147,
       0.96586722, 0.96673227, 0.96756091, 0.968377 , 0.96918583])
```

Zaman Değişkeni

TransactionDT: belirli bir referans zamana olan uzaklık

- Saat
- Gün
- Ay
- Haftanın günü
- Tarih

Saat değişkeni 4 kategoriye ayrıldı



- high(6,7,8,9) 27198
- medium(3,4,5,10,11) 65375
- low(11,12,16-2) 846167
- very low(13,14,15) 158491

İşlem Yapılan Cihaza İlişkin Özellik Türetme

```
dataframe['device_name'] = dataframe['DeviceInfo'].str.split('/', expand=True)[0]
dataframe['device_version'] = dataframe['DeviceInfo'].str.split('/', expand=True)[1]
dataframe['OS_id_30'] = dataframe['id_30'].str.split(' ', expand=True)[0]
dataframe['version_id_30'] = dataframe['id_30'].str.split(' ', expand=True)[1]
dataframe['browser_id_31'] = dataframe['id_31'].str.split(' ', expand=True)[0]
dataframe['version_id_31'] = dataframe['id_31'].str.split(' ', expand=True)[1]
```

- Cihaz markası
- Version
- İşletim Sistemi
- Browser

Telefon markaları

⚠ 39 ⚡ 264 ✅ 103 ^

```
dataframe.loc[dataframe['device_name'].str.contains('SM', na=False), 'device_name'] = 'Samsung'
dataframe.loc[dataframe['device_name'].str.contains('SAMSUNG', na=False), 'device_name'] = 'Samsung'
dataframe.loc[dataframe['device_name'].str.contains('GT-', na=False), 'device_name'] = 'Samsung'
dataframe.loc[dataframe['device_name'].str.contains('Moto G', na=False), 'device_name'] = 'Motorola'
dataframe.loc[dataframe['device_name'].str.contains('Moto', na=False), 'device_name'] = 'Motorola'
dataframe.loc[dataframe['device_name'].str.contains('moto', na=False), 'device_name'] = 'Motorola'
dataframe.loc[dataframe['device_name'].str.contains('LG-', na=False), 'device_name'] = 'LG'
dataframe.loc[dataframe['device_name'].str.contains('rv:', na=False), 'device_name'] = 'RV'
dataframe.loc[dataframe['device_name'].str.contains('HUAWEI', na=False), 'device_name'] = 'Huawei'
dataframe.loc[dataframe['device_name'].str.contains('ALE-', na=False), 'device_name'] = 'Huawei'
dataframe.loc[dataframe['device_name'].str.contains('-L', na=False), 'device_name'] = 'Huawei'
dataframe.loc[dataframe['device_name'].str.contains('Blade', na=False), 'device_name'] = 'ZTE'
dataframe.loc[dataframe['device_name'].str.contains('BLADE', na=False), 'device_name'] = 'ZTE'
dataframe.loc[dataframe['device_name'].str.contains('Linux', na=False), 'device_name'] = 'Linux'
dataframe.loc[dataframe['device_name'].str.contains('XT', na=False), 'device_name'] = 'Sony'
dataframe.loc[dataframe['device_name'].str.contains('HTC', na=False), 'device_name'] = 'HTC'
dataframe.loc[dataframe['device_name'].str.contains('ASUS', na=False), 'device_name'] = 'Asus'
dataframe.loc[dataframe.device_name.isin(dataframe.device_name.value_counts()[dataframe.device_name.value_counts() < 200].index), 'device_name'] = "Others"
```

E-mail Domain Değişkenleri

E-mail check: Müşteri ve parayı alanın domainleri aynı mı?

E-mail uzantısı: com, us, mx, es, fr, de, uk, jp...

E-mail sağlayıcı: gmail, yahoo, hotmail, comcast, outlook, icloud, att, msn, live...

Model

- Logistic Regression
- Random Forest
- XGBoost
- Light GBM

XGB

Accuracy: 0.6279

Auc: 0.7646

Recall: 0.5055

Precision: 0.3981

F1: 0.137

LightGBM

Accuracy: 0.6889

Auc: 0.7853

Recall: 0.4982

Precision: 0.3969

F1: 0.1613

LR

Accuracy: 0.9652

Auc: 0.7134

Recall: 0.0234

Precision: 0.5776

F1: 0.0439

RF

Accuracy: 0.8194

Auc: 0.7624

Recall: 0.4109

Precision: 0.5504

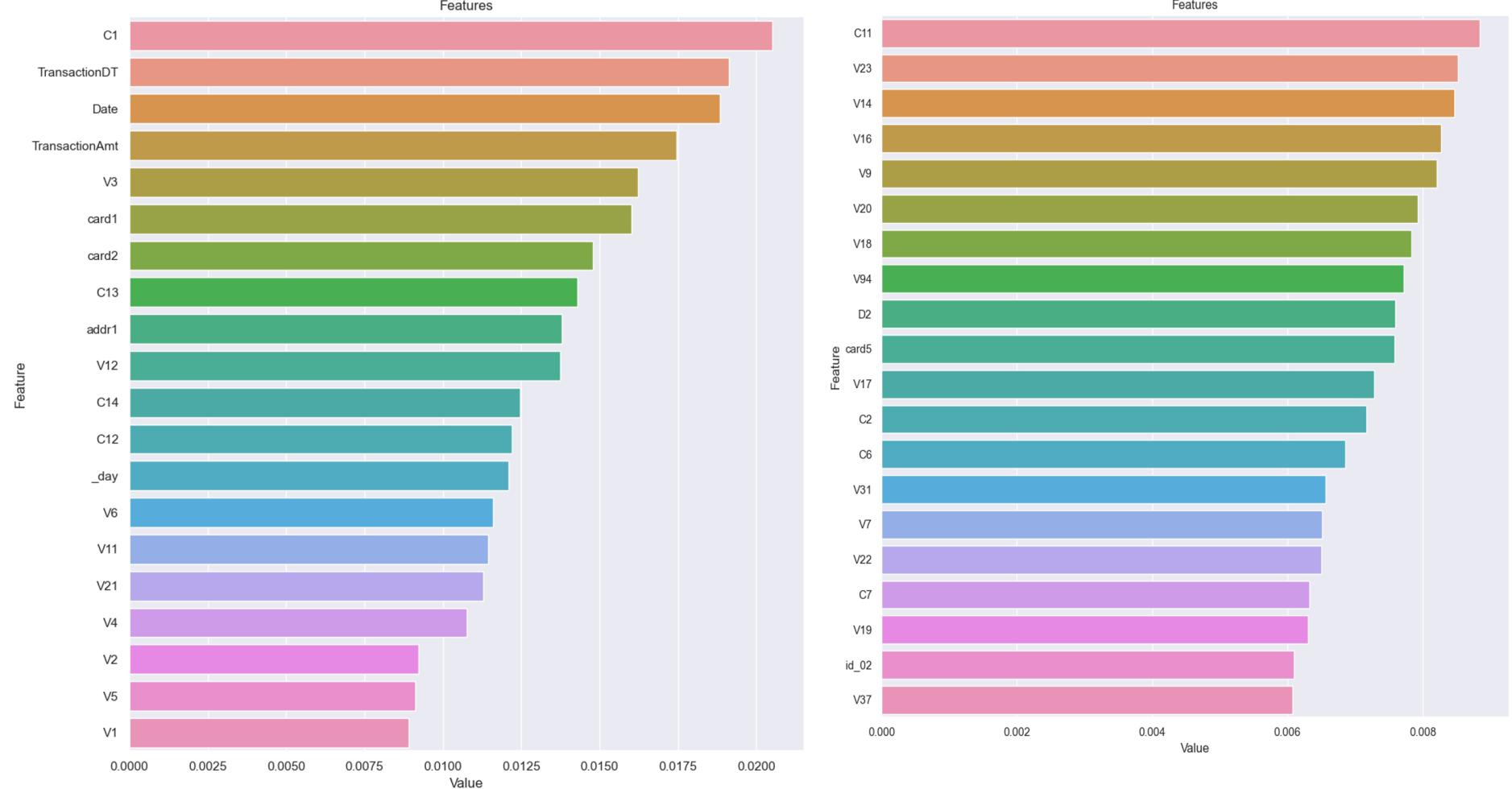
F1: 0.3163

```
##### XGB #####
Accuracy: 0.637
Auc: 0.7715
Recall: 0.4707
Precision: 0.4123
F1: 0.1067
```

```
##### XGB #####
Accuracy: 0.67
Auc: 0.79
Recall: 0.5
Precision: 0.41
F1: 0.16
#####
LightGBM #####
Accuracy: 0.7
Auc: 0.79
Recall: 0.49
Precision: 0.41
F1: 0.17
```

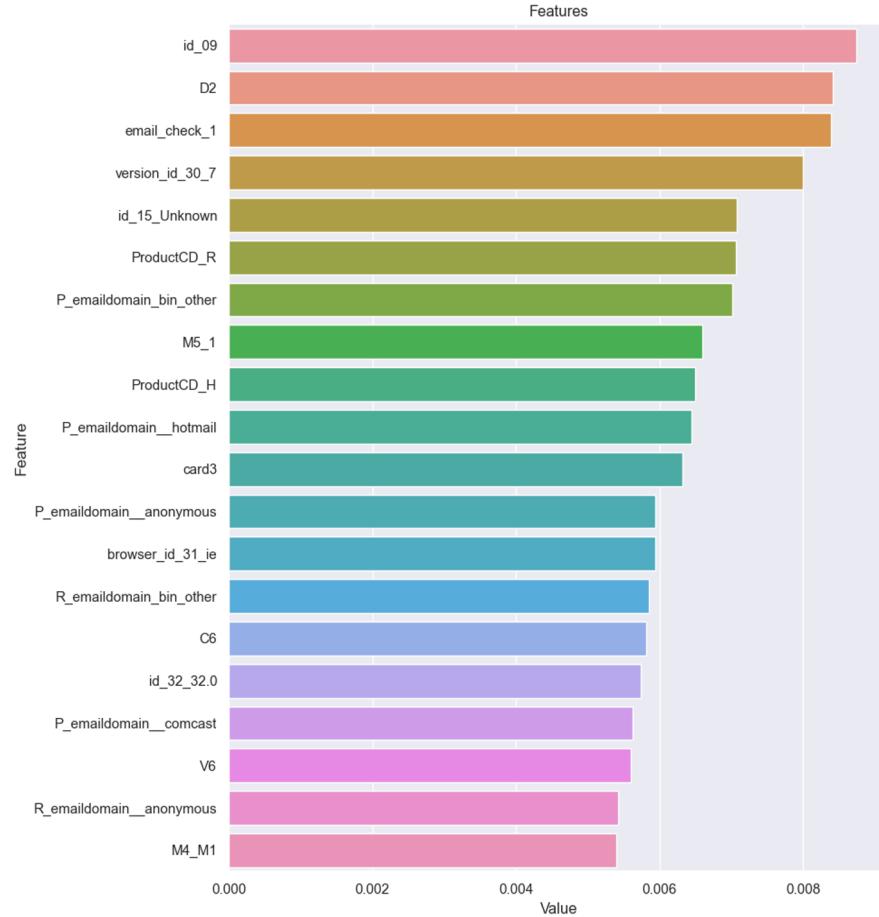
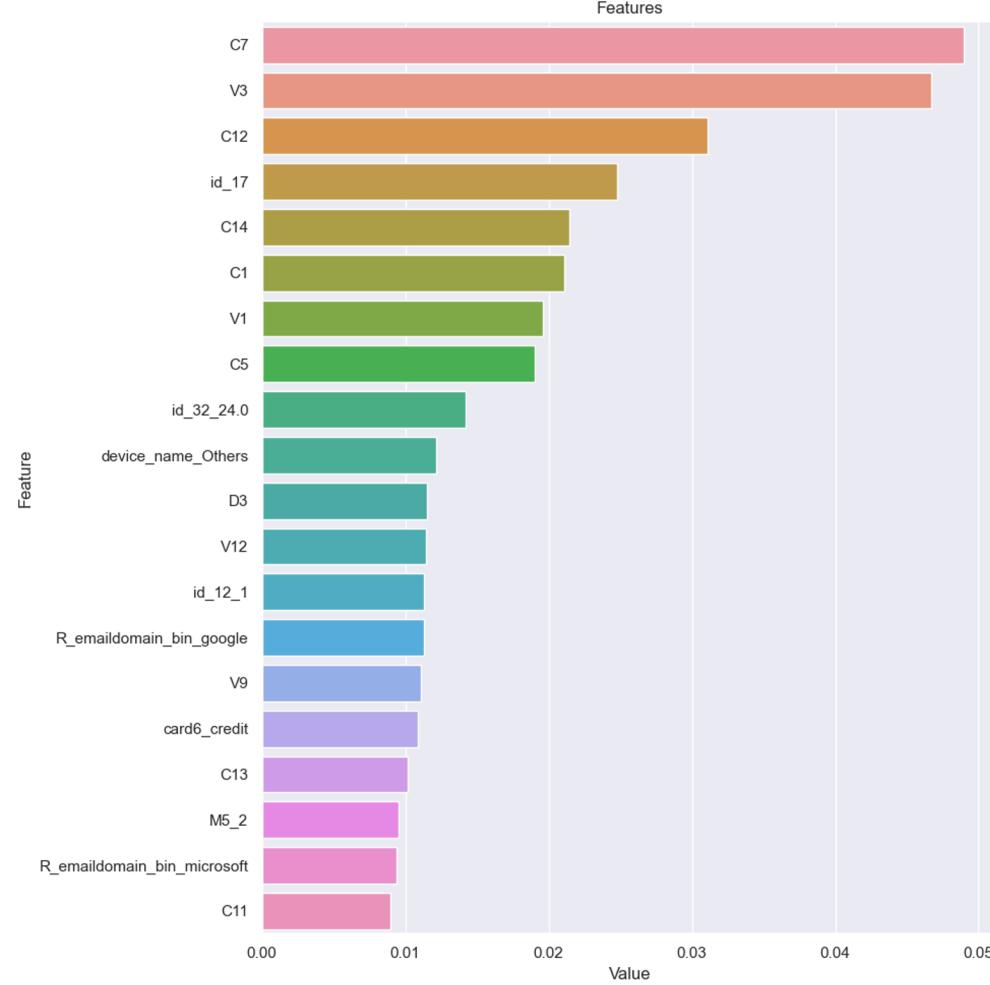
Değişken Önem Düzeyi

RANDOM FOREST



Değişken önem düzeyi

XGBOOST

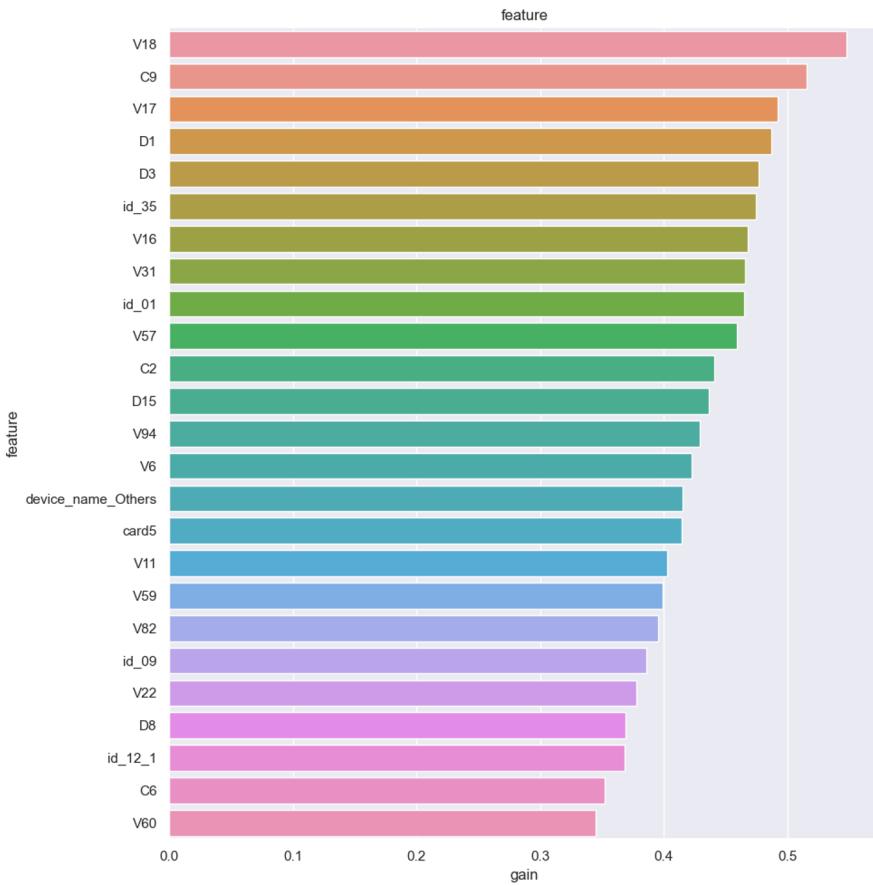
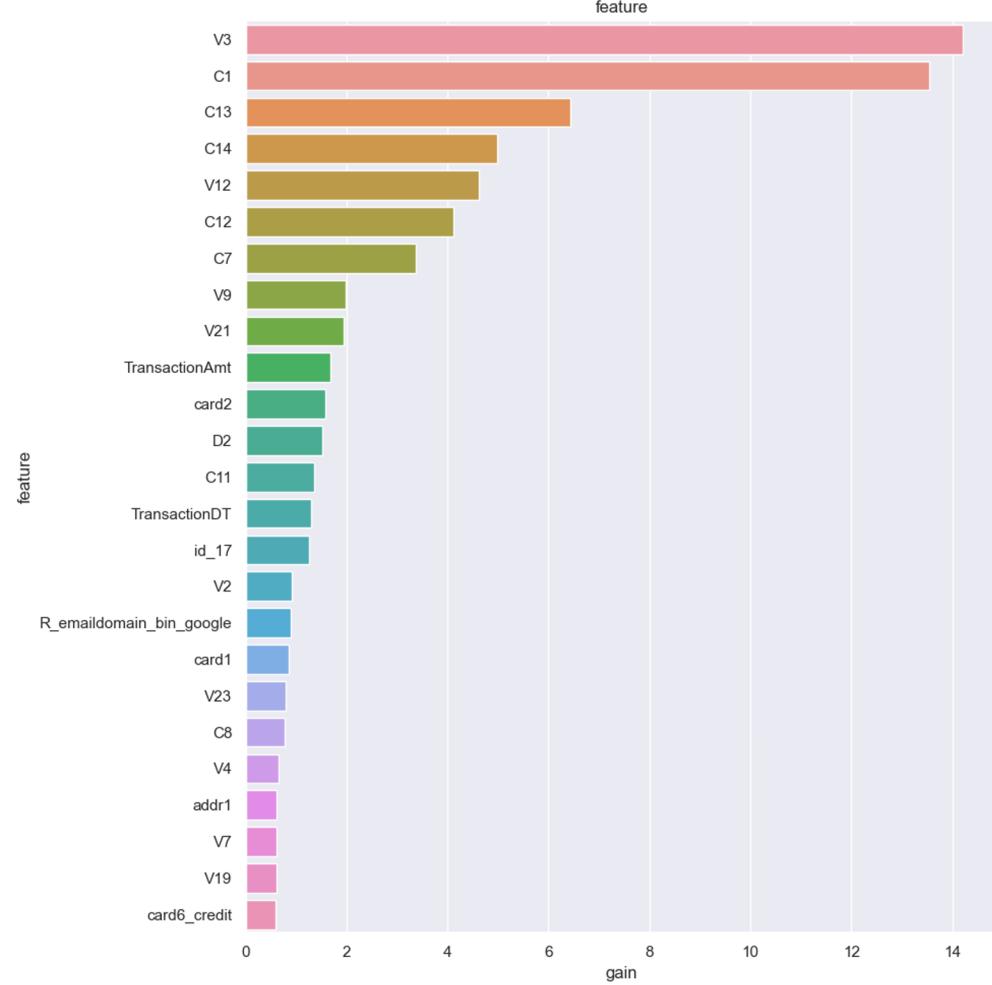


Gain

		feature	split	gain
2		V3	37	14.203
109		C1	125	13.545
120		C13	97	6.441
121		C14	68	4.981
11		V12	41	4.623
..	
197	P_emaildomain_bin_apple		0	0.000
195	id_34_match_status2		0	0.000
194	id_34_match_status1		0	0.000
190	id_16_2		0	0.000
277	_hour_Very_low		0	0.000

[278 rows x 3 columns]

- Modelimiz 278 değişken üzerine kuruldu
- 59 değişkenin gain skoru 0
- Modele katkısı olan değişken sayısı 219



Hiperparametre Optimizasyonu

```
lgbm_params = {"learning_rate": [0.01, 0.1],  
                "n_estimators": [100, 500],  
                "colsample_bytree": [0.5, 1]}
```

```
##### LightGBM #####  
Accuracy: 0.9693  
Auc: 0.8035  
Recall: 0.1406  
Precision: 0.9194  
F1: 0.2323
```

KAGGLE

Leaderboard

[Raw Data](#)[Refresh](#)

YOUR RECENT SUBMISSION

**submission.csv**

Submitted by guldane durukan · Submitted just now

Score: 0.839117

Public score: 0.880260

[↓ Jump to your leaderboard position](#)