BURSA TEKNİK ÜNİVERSİTESİ

Mühendislik ve Doğa Bilimleri Fakültesi – Bilgisayar Mühendisliği Bölümü

BLM0463_Veri Madenciliğine Giriş

Bahar 2022

Cam Sınıflandırma

VERİ MADENCİLİĞİ DERSİ PROJESİ

MUSTAFA EREN
(18360859024)

11.06.2022

İÇİNDEKİLER

Sayfa

ŞEKİL LİSTESİ

Sayfa

1. GİRİŞ

Bu yazıda kullanılan veri seti, UCI Machine Learning Repository'den alınan “Glass Identification Dataset’tir. Cam sınıflandırma probleminin incelenmesi, suç mahallinde bırakılan cam, doğru teşhis edildiği takdirde delil olarak kullanılabilir. Sık görülen bir vaka çalışması gereksinimi, bir suç mahallindeki camın bir şüpheliyle ilişkili olduğu bulunan cam parçacıklarıyla karşılaştırılmasıdır. Bu tür cam parçacıkları genellikle aşırı derecede küçüktür. Adli bağlamda önemli olabilecek bu küçük cam parçalarının belirlenmesi ve karşılaştırılması önemlidir. Ana amaç, ölçülen ana bileşene dayalı olarak tek bir cam parçasını doğru şekilde sınıflandırmaktır.

2. METODOLOJİ

Cam sınıflandırma problemini ele almak için Decision Tree Method kullanılmıştır. Bu veri madenciliği uygulaması için Python programlama dili ve kütüphaneleri kullanılmıştır. Bu çalışma veri analizi, ön işleme ve modelleme aşamalarından geçer. Veri analizi ve ön işleme adımları, modelleme adımları için hazılırlığı amaçlar. Modelleme aşaması, cam tanımlama problemi için yüksek doğrulukta bir tahmin modeli oluşturmak için Decision Tree Method kullanmayı içerir.

2.1 Veri Analizi

UCI Machine Learning Repository, birçok veri kümesi sağlar. Bunlardan biri, bileşenlerine göre camın türünü belirleyen cam tanımlama veri setidir. Bu veri kümesindeki Örnek sayısı 214'tür. Bu veri kümesi, tabloda gösterildiği gibi 10 özniteliğe sahiptir.

Attribute Name	Attribute Type
Id	Sayısal
Kırılma Indeksi	Sayısal
Sodyum	Sayısal
Magnezyum	Sayısal
Alüminyum	Sayısal
Potasyum	Sayısal

Silikon	Sayısal
Kalsiyum	Sayısal
Baryum	Sayısal
Demir	Sayısal
Tip	Kategorik

2'den 9'a kadar olan nitelikler, karşılık gelen oksitte ağırlık yüzdesi kullanılarak ölçülür. Camın, cam işlemeli camlar, cam işlemesiz camlar, cam işlemeli camlar, cam işlemesiz camlar, kaplar, sofra takımları ve farlar gibi farklı kullanımları tanımlayan yedi türü vardır. Sonuç olarak, her tip 1'den 7'ye kadar bir sayı ile temsil edilir.

```
>>> df.head(5)
   Id      RI     Na    Mg    Al     Si     K    Ca   Ba   Fe  Type
0   1  1.52101  13.64  4.49  1.10  71.78  0.06  8.75  0.0  0.0     1
1   2  1.51761  13.89  3.60  1.36  72.73  0.48  7.83  0.0  0.0     1
2   3  1.51618  13.53  3.55  1.54  72.99  0.39  7.78  0.0  0.0     1
3   4  1.51766  13.21  3.69  1.29  72.61  0.57  8.22  0.0  0.0     1
4   5  1.51742  13.27  3.62  1.24  73.08  0.55  8.07  0.0  0.0     1
>>>
```

Şekil 1 : Veri setinin ilk 5 değeri

```
RangeIndex: 214 entries, 0 to 213
Data columns (total 11 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Id      214 non-null    int64
 1   RI      214 non-null    float64
 2   Na      214 non-null    float64
 3   Mg      214 non-null    float64
 4   Al      214 non-null    float64
 5   Si      214 non-null    float64
 6   K       214 non-null    float64
 7   Ca      214 non-null    float64
 8   Ba      214 non-null    float64
 9   Fe      214 non-null    float64
 10  Type    214 non-null    int64
dtypes: float64(9), int64(2)
```

Şekil 2 : Veri setinin tipleri

Veri tiplerinin sayısal olduğu görülebilir. Eksik değer bulunmamaktadır.

	Id	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type
count	214.000000	214.000000	214.000000	214.000000	214.000000	214.000000	214.000000	214.000000	214.000000	214.000000	214.000000
mean	107.500000	1.518365	13.407850	2.684533	1.444907	72.650935	0.497056	8.956963	0.175047	0.057009	2.780374
std	61.920648	0.003037	0.816604	1.442408	0.499270	0.774546	0.652192	1.423153	0.497219	0.097439	2.103739
min	1.000000	1.511150	10.730000	0.000000	0.290000	69.810000	0.000000	5.430000	0.000000	0.000000	1.000000
25%	54.250000	1.516522	12.907500	2.115000	1.190000	72.280000	0.122500	8.240000	0.000000	0.000000	1.000000
50%	107.500000	1.517680	13.300000	3.480000	1.360000	72.790000	0.555000	8.600000	0.000000	0.000000	2.000000
75%	160.750000	1.519157	13.825000	3.600000	1.630000	73.087500	0.610000	9.172500	0.000000	0.100000	3.000000
max	214.000000	1.533930	17.380000	4.490000	3.500000	75.410000	6.210000	16.190000	3.150000	0.510000	7.000000

Şekil 3 : Veri setinin istatistikleri

Veri setinin betimsel istatistikleri görüntülenmiştir. Ortalama, standart sapma, minimum, maksimum değerler gibi hesaplamaları göstermektedir.

	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type
RI	1.000000	-0.191885	-0.122274	-0.407326	-0.542052	-0.289833	0.810403	-0.000386	0.143010	-0.164237
Na	-0.191885	1.000000	-0.273732	0.156794	-0.069809	-0.266087	-0.275442	0.326603	-0.241346	0.502898
Mg	-0.122274	-0.273732	1.000000	-0.481799	-0.165927	0.005396	-0.443750	-0.492262	0.083060	-0.744993
Al	-0.407326	0.156794	-0.481799	1.000000	-0.005524	0.325958	-0.259592	0.479404	-0.074402	0.598829
Si	-0.542052	-0.069809	-0.165927	-0.005524	1.000000	-0.193331	-0.208732	-0.102151	-0.094201	0.151565
K	-0.289833	-0.266087	0.005396	0.325958	-0.193331	1.000000	-0.317836	-0.042618	-0.007719	-0.010054
Ca	0.810403	-0.275442	-0.443750	-0.259592	-0.208732	-0.317836	1.000000	-0.112841	0.124968	0.000952
Ba	-0.000386	0.326603	-0.492262	0.479404	-0.102151	-0.042618	-0.112841	1.000000	-0.058692	0.575161
Fe	0.143010	-0.241346	0.083060	-0.074402	-0.094201	-0.007719	0.124968	-0.058692	1.000000	-0.188278
Type	-0.164237	0.502898	-0.744993	0.598829	0.151565	-0.010054	0.000952	0.575161	-0.188278	1.000000

Şekil 4 : Sayısal Değişkenlerin Korelasyon Matrisi

Korelasyon, olasılık kuramı ve istatistikte iki rassal değişken arasındaki doğrusal ilişkinin yönünü ve gücünü belirtir. Şekil 4'de sayısal değişkenlerin korelasyonunu ifade etmektedir.

2.2 Veri Ön İşleme

Bir veri kümesi doğru formatta teslim edilse bile, veri madenciliği algoritmasını uygulayabilmek ve analiz sonucunun kalitesini artırmak için yine de ön işlemeye ihtiyaç duyabilir.
Birçok veri ön işleme tekniği vardır. Bu çalışmada kullanılan teknikler eksik değer analizi, aykırı değer analizi, oversampling işlemleri kullanılmıştır.

2.2.1 Eksik Değer Analizi

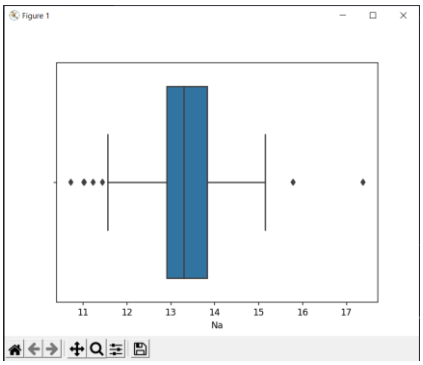
Verilerde eksik değer bulunmamakta. O yüzden bu adım atlanabilir. Eğer eksik değerler olsaydı o satır düşürülebilirdi. Eğer eksik veri sayısı fazla ise eksik değerler ortalama değerlerle doldurulabilir.

```
>>> df.isnull().sum()  # null olan değerlerin sayısı 0'dır yani bu adımı geçebiliriz
Id      0
RI      0
Na      0
Mg      0
Al      0
Si      0
K       0
Ca      0
Ba      0
Fe      0
Type    0
dtype: int64
>>>
```

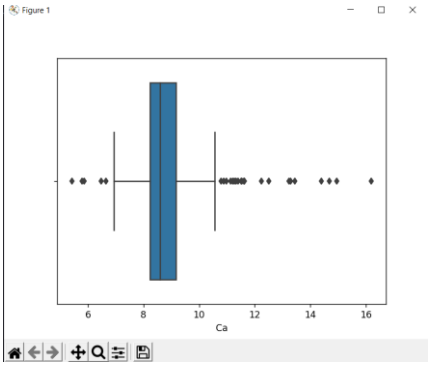
Şekil 5 : Eksik Değerlerin Sayısı

2.2.2 Aykırı Değer Analizi

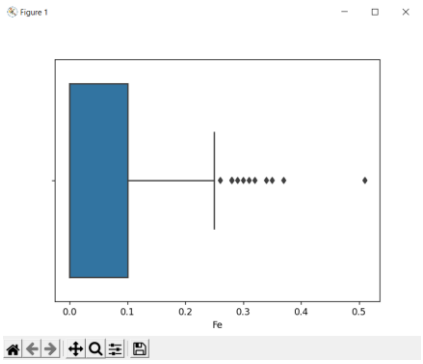
Verilerde aykırı değerler mevcuttur. Bunları boxplot grafiğinde daha iyi görebiliriz. Görselde Na değişkeninin aykırı değerlere sahip olduğunu görebiliriz.



Şekil 6 : Na Değişkeninin BoxPlot grafiği ve aykırı değerlere sahip olduğunun gösterilmesi

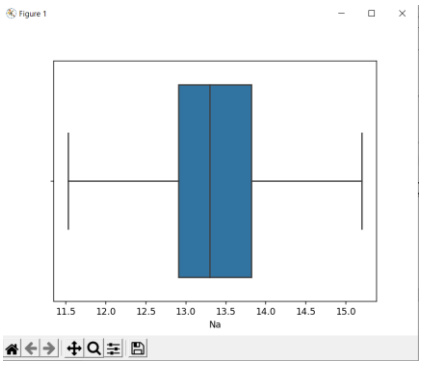


Şekil 7 : Ca Değişkeninin BoxPlot grafiği ve aykırı değerlere sahip olduğunun gösterilmesi

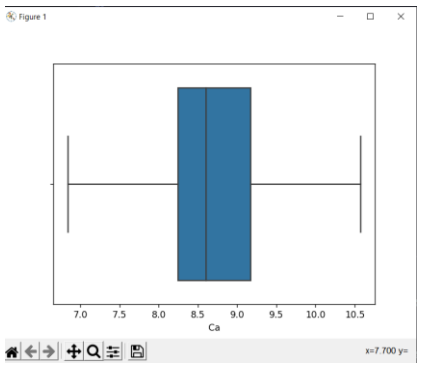


Şekil 8 : Fe Değişkeninin BoxPlot grafiği ve aykırı değerlere sahip olduğunun gösterilmesi

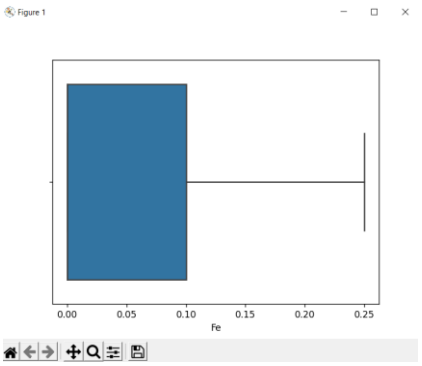
Aykırı verilerin baskılanması gerekemekte. Baskılama projede yazılan fonksiyonlar aracılığıyla sağlanır. Parametre olarak verilen değerler altında yer alanlar alt sınıra, üst sınırın üstünde yer alan değerler ise üst sınıra baskılanır. Aykırı değerlerin baskılanmasından sonra Na değişkeninin boxplot grafiği şu şekildedir.



Şekil 9 : Na Değişkenin yeni BoxPlot grafiği ve aykırı değerlerin baskılandığının gösterilmesi



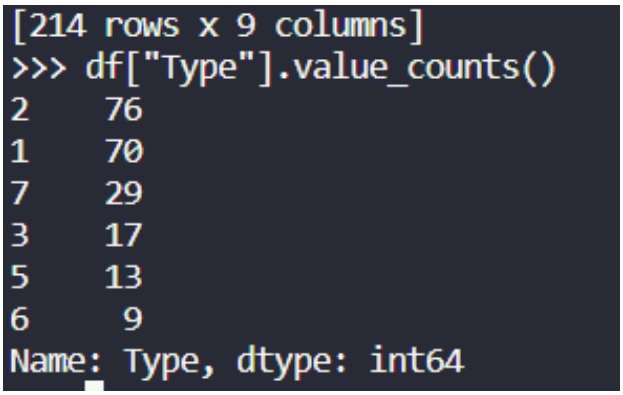
Şekil 10 : Ca Değişkenin yeni BoxPlot grafiği ve aykırı değerlerin baskılandığının gösterilmesi



Şekil 11 : Fe Değişkenin yeni BoxPlot grafiği ve aykırı değerlerin baskılandığının gösterilmesi

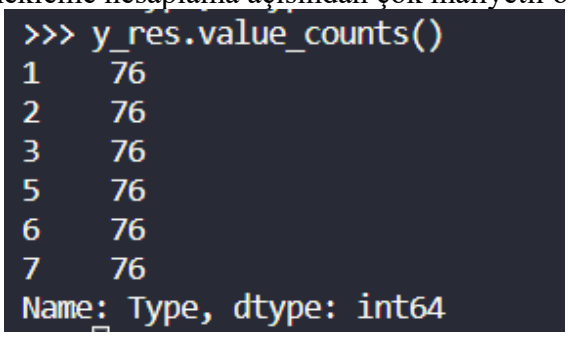
2.2.3 Oversampling İşlemi

Dengesiz veri problemi olarak bilinen sınıf dengesizliği, veri bilimi projeleri çerçevesinde dikkat edilmesi gereken bir sorundur. Sınıflandırma algoritmalarının çoğu, eğitim setlerinin iyi dengelendiğini varsayar. Algoritmaların amacı, genellikle, doğru tahmin oranını maksimize etmektir. Ancak Cam Sınıflandırma veri setinde veriler eşit dağılmamıştır. Sınıfların sayısı görselde gösterilmiştir.



Şekil 12 : Dengesiz sınıf dengesizliğinin gösterilmesi

Etiket bilgisi az olan örnekler için model yeterince eğitilmediği için modelin bu grup için hatalı tahminler yapması muhtemeldir. Aşırı örnekleme, eşit sınıf dağılımları elde edilene kadar azınlık sınıfının örneklerini çoğaltır. Bu konudaki yöntemlerinin çoğu, azınlık sınıfının örneklerini kopyaladığından, aşırı öğrenme(overfiting) olma olasılığı artar. Ayrıca, yüksek düzeyde dengesiz dağılıma sahip büyük bir veri kümesi olması durumunda, aşırı örnekleme hesaplama açısından çok maliyetli olabilir.



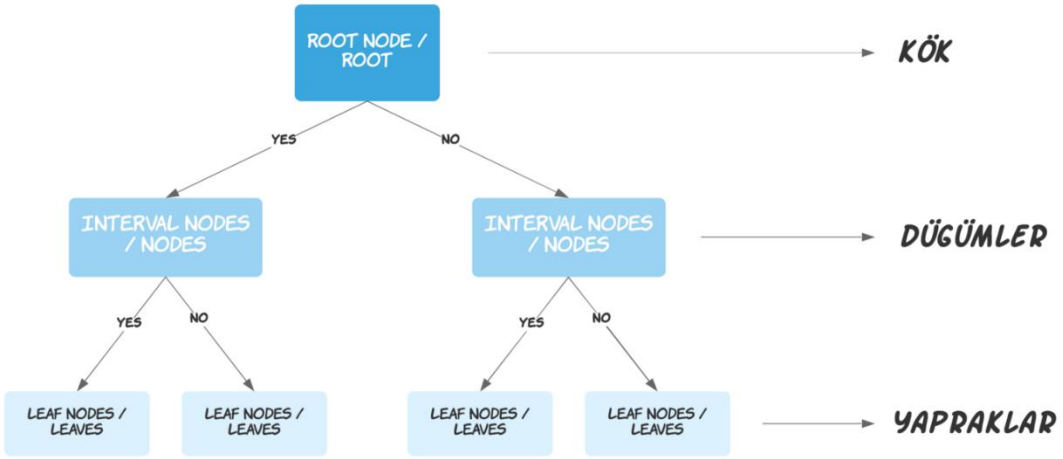
Şekil 13 : Oversampling uyguladıktan sonra sınıfların içerdiği veri sayısı.

3. MODELLEME

Veri analizi ve ön işleme adımlarını tamamladıktan sonra modelleme aşamasına geçilir. Modelleme için Decision Tree Method kullanılmıştır.

3.1 Desicion Tree Method

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



Şekil 14 : Karar Ağacı

3.1.1 ID3 Algoritması

J.R. Quinlan, tarafından 1986 yılında bir veri setinden "karar ağacı" üretmek için geliştirilen ID3 algoritması geliştirmiştir. Bu algoritma aşağıdan yukarı (top-down : kökten alt dallara doğru) ve greedy search (sonuca en yakın durum) teknikleri kullanılır. Decision Tree konusunda sıklıkla göreceğiz C4.5 algoritması ID3 algoritmasının bir uzantısıdır. ID3 algoritması Entropy ve Information Gain üzerine inşa edilmiştir.

3.1.1.1 Entropy

Rastgeleliğe, belirsizliği ve beklenmeyen durumun ortaya çıkma olasılığını gösterir. Eğer örnekler tamamı düzenli / homojen ise entropisi sıfır olur. Eğer değerler birbirine eşit ise entropi 1 olur. Örneğin Futbol Oyna hepsi "Evet" veya "Hayır" olsa entropi sıfır olurdu.

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Şekil 15 : Entropi formülasyonu

3.1.1.2 Information Gain

Bilgi kazanımı, bir veri setini bir özellik üzerinde böldükten (Örneğin E(FutbolOyna, HavaDurumu)) sonra tüm entropiden (E(FutbolOyna)) çıkarmaya dayanır. Entropinin küçük değer içermesi durumunda özelliğin önemi Decision Tree algoritması ID3 için artmaktadır. Diğer taraftan 1'e yaklaştıkça özelliğinin önemi azalır. Ancak information gain'de olay tam tersidir ve bu açıdan entropinin tersi gibi düşünülebilir. Decision Tree inşa edilirken en yüksek değerleri information gain'e sahip özellik seçilir.

$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

Şekil 16 : Information Gain formülasyonu

4. VERİ MODELLEME SONUÇLARI

4.1 Parametre Optimizasyonu Öncesi Sonuçlar

Metrik	Accuracy	Recall	Precision	F1 Score
Sonuç	0.8913	0.8913	0.8987	0.8916

Varsayılan parametrelerle eğitilen modelin sonucu yukarıdaki tabloda gösterilmiştir. Bu sonuçlar sadece varsayılan parametrelerle değil train ve test kümesinin rastgele olarak seçildiği bir modelin sonuçlarıdır.

4.2 Parametre Optimizasyonu ve K-Fold Uygulanması Sonrası Sonuçlar

Metrik	Accuracy	Recall	Precision	F1 Score
Sonuç	0.8804	0.8804	0.8879	0.8791

Sonuçlarda değerlerin düştüğü gözlemlenmiştir. Bir önceki base modelin rastgelelikten etkilenmesindendir. Cross-Validation(K-fold) bu rastgeleliği ortadan kaldırır. Bu değerin düşmesi kötü gibi görünebilir ancak sonuçlar daha doğrudur. Rastgelelikten etkilenen bir modelin gerçek verilerde hata payı daha yüksektir. K-fold ile bunun önüne geçmek hedeflenmiştir.

4.3 Sonuçların Karşılaştırılması

4.3.1 KNN Algoritması ile Glass Identification

Mashael S. Aldayel tarafından yazılmış 'K-Nearest Neighbor Classification for Glass Identification Problem' isimli makale de sınıflandırma için KNN algoritması kullanılmıştır.

TABLE IV. CLASSIFICATION ACCURACY DETAILS IN GLASS DATASET

Method	Precision	Recall	F-Measure	Accuracy Rate
HNB	0.801	0.799	0.797	79.9065%
KNN (K=1)	0.796	0.79	0.789	78.972%
KNN (K=3)	0.789	0.78	0.778	78.0374%
KNN (K=5)	0.768	0.748	0.745	74.7664%
KNN (K=7)	0.756	0.738	0.733	73.8318 %
KNN (K=9)	0.707	0.701	0.685	70.0935%
Proposed Voting, K =1 (KNN+HNB)	0.806	0.804	0.802	80.3738%

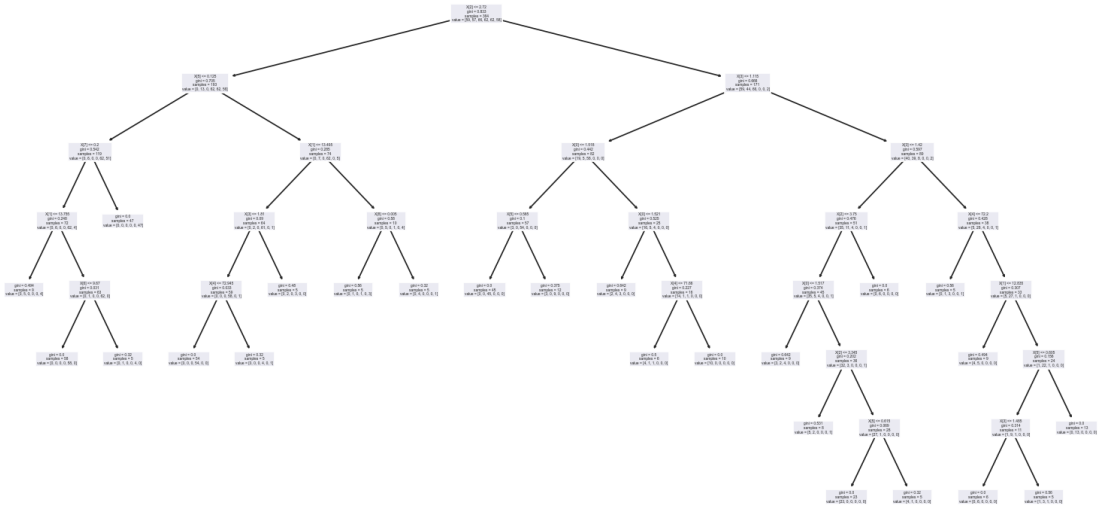
Şekil 17 : Glass Identification Dataset üzerinde eğittiği modeldeki başarı oranları tabloda gösterilmiştir.

Makale de KNN algoritmasındaki en yakın komşu sayısını artırdığında başarının bir miktar artığı gözlenmiştir.

4.3.2 Artificial Neural Network ile Glass Identification

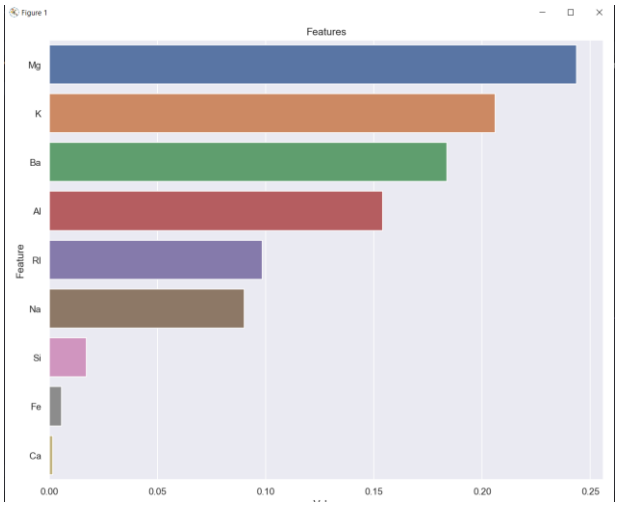
El-Khatib, Abu-Nasser, Abu-Naser (2019)'in "Glass Classification Using Artificial Neural Network" isimli makale de Accuracy değeri 96.70% olarak gözlenmiştir.

4.4 Eğitilen Ağacın Görselleştirilmesi



Şekil 18 : Eğitilen Ağacın Görselleştirilmesi

4.5 Model Değişkenlerinin Önem Düzeyleri



Şekil 19 : Değişkenlerin Önem Düzeyleri

Bu analizin sonucu, hedef değişkenimizi bulma da Magnezyum, Kalsiyum, Baryum özniteliklerinin diğerlerinden daha fazla önem düzeyine sahip olduğunu göstermektedir.

5. SONUÇ

Suç delili verilerinden elde edilen bilgilerin keşfi, etkili kriminolojik soruşturma yapabilmek için önemlidir. Veri madenciliğinin amacı, verilerden bilgi çıkarmak ve açık ve makul modeller üretmektir. Bu yazıda cam sınıfını tanımlamada Decision Tree Method uygulanmıştır. Bazı sınıflardaki veri diğerlerine göre çok az olduğun oversampling uygulanmıştır ve her sınıftan 76 veri ile model eğitilmiştir. Cross-Validation uygulanmasıyla beraber Accuracy değeri 0.8804 olduğu görülmüştür.

KAYNAKLAR

- **El-Khatib, Abu-Nasser, Abu-Naser (2019)**
- **K-Nearest Neighbor Classification for Glass Identification Problem Mashael Dayel (2013)**
- **https://erdincuzun.com/makine_ogrenmesi/decision-tree-karar-agaci-id3-algoritmasi-classification-siniflama/**
- **https://medium.com/bili%C5%9Fim-hareketi/cross-validation-nedir-nas%C4%B1l-%C3%A7al%C4%B1%C5%9F%C4%B1r-4ec4736e5142**
- **https://veribilimcisi.com/2017/07/18/karisiklik-matrisi-nedir/**
- **https://www.veribilimiokulu.com/dengesiz-veri-setlerinde-modelleme/**
- **https://devreyakan.com/performans-metrikleri/**