<Örüntü Tanıma Raporu>

**24 May 2022**

Prepared by <Hüseyin Oral 170101033>

# Introduction

## Background

Projenin Amacı: Resimlerin, black(siyah), grizzly(boz), teddy(oyuncak) ayı kategorilerine göre sınıflandırılması. Projenin oluşturulma sebebi ise akademik çalışma.

## Objectives

* Veri seti işlenmesini öğrenmek.
* Veri setlerinde boyut indirgeme ve gürültü temizlemek.
* Veri seti için doğru sınıflandırma algoritmasını bulabilmek.

# Methodology

## Research Questions

1. Hangi sınıflandırma algoritması bu veri seti için daha iyi sonuç verdi ?
2. Sınıflandırma algoritmalarının parametlerinin değişimi sonuçları etkiledimi ?
3. Resimlerde gürültü temizlemek işe yaradımı ?
4. Veri üzerinde boyut indirgeme ne gibi bir etki yarattı ?

## Sample

Veri seti Kaggle ‘dan alınmıştır. Projedeki resim sayısı black kategorisinde 96, grizzly kategorisinde 98, teddy kategorisinde 94.Toplam 288 tane resim vardır. Veri seti sınıflandırma yapılabilecek veri setleri içerisinden seçilmiştir.

## Data Collection

Veriler Kaggle sitesine Anirudh Gokulaprasad tarafından yüklenmiştir. Veriler 3 kategorideki ayı türlerinin fotoğrafları çekilerek toplanmıştır. Veriler .jpg formatındadır.

## Data Analysis

Veriler .jpg formatında indirildi. Python kullanarak işlendi. Veriler resim olduğu için boyutları 32x32 olarak işlendi.

## Limitations

Toplam 3 kategori olup bunların toplam resim sayısı 288 dir bu sayı sınıflandırma için çok iyi sonuç ortaya çıkarmayabilir.

## Research Design

|  |  |
| --- | --- |
| Research Question | Method Used to Answer Question |
| Hangi sınıflandırma algoritmaları kullanıldı ? | KNN, Random Forest, Naïve Bayes , SVC |
| KNN için hangi komşu sayıları kullanıldı ? | 5,15 ve 30. |
| KNN için hangi uzaklık hesaplama fonksiyonu kullanıldı ? | Minkowski |
| Random Forest için kullanılan ağaç sayıları ? | 5, 25 ve 50. |
| Gürültü temizlemek için hangi fonksiyon kullanıldı ? | OpenCV’ den fastNlMeansDenoising kullanıldı. |
| Boyut indirgeme yöntemi için ne kullanıldı ? | PCA(Principal Component Analysis) |
| Hangi çapraz doğrulama tekniği kullanıldı ? | K-Fold (K katmanlı Çapraz Doğrulama Kullanılmıştır ). K değeri 5 alınmıştır. |
| Sınıflandırma sonuçları neye göre karşılaştırlıyıor ? | Sınıflandırma sonuçları Roc\_auc değerlerine göre belirleniyior. |

# Results

## Sınıflandırma algoritmalarının çeşitleri ve parametrelere göre sonuçları :

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sınıflandırma Algoritmaları** | **Parametre** | **Accuracy** | **F1-Score** | **Recall** | **Precision** | **Roc\_Auc** |
| KNeighborsClassifier | K=5 | 0.76158 | 0.76855 | 0.76158 | 0.83592 | 0.85595 |
| KNeighborsClassifier | K=15 | 0.74000 | 0.74264 | 0.74000 | 0.77903 | 0.87143 |
| KNeighborsClassifier | K=30 | 0.63526 | 0.65011 | 0.63526 | 0.75683 | 0.82505 |
| Naïve Bayes | --- | 0.48842 | 0.49766 | 0.48842 | 0.53557 | 0.74681 |
| Random Forest | N=5 | 0.62474 | 0.63846 | 0.68789 | 0.68391 | 0.79930 |
| Random Forest | N=25 | 0.68789 | 0.71627 | 0.73947 | 0.71867 | 0.87224 |
| Random Forest | N=50 | 0.69789 | 0.68977 | 0.73947 | 0.72499 | 0.87638 |
| SVC | --- | 0.69789 | 0.64527 | 0.69789 | 0.75696 | 0.87994 |

K = Komşu sayısı.

N=Oluşturulan ağaç sayısı.

# Discussion

Projede sınıflandırma algoritmaları ve parametrelerin etkisi 3.1 ‘ deki tabloda gözükmektedir. Sonuçlar değerlendirilmesi Roc\_Auc değerine göre yapılmıştır. KNN(KNeighborsClassifier) sınıflandırma algoritmasında komşu sayısı parametresi (K) ‘nin değerleri değiştirilerek sonuçlar incelendiğinde her zaman komşu sayısının fazla olması daha iyi sonuç vermedeği ortaya çıkmıştır K=5 iken Roc\_Auc değeri 0.85595, K=15 iken 0.087143 ve K=30 iken 0.82505 değerlerine sahiptir. Bu sonuçlar karşılaştırıldığında en iyi sonucun ne en fazla komşu sayısı yada en az komşu sayısında olduğu ortaya çıkmıştır. K parametresinin değeri rastgele verildiği için en iyi sonuç deneyerek bulmaktadır. Burda en iyi sonuç K=15 değerinde ortaya çıkmıştır. Random forest ‘de ise ağaç sayısı arttıkça sonucun daha iyi olduğu gözlenmiştir N=5 iken 0.7993, N=25 iken 0.87224 ve N=50 iken 0.87638 olduğundan ağaç sayısı arttıkça sonuç daha iyiye gitmiştir.Naive Bayes ve SVC’ de ise K veya N gibi bir parameter olmadığı için sonuçları sırasıyla 0.74681 ve 0.87994‘ dür. Bu sonuçlar sıralandığında en iyi sonuçtan itibaren sıra şöyledir :

1. SVC
2. Random Forest (N=50)
3. Random Forest (N=25)
4. KNN (K=15)
5. KNN (K=5)
6. KNN (K=30)
7. Random Forest (N=5)
8. Naive Bayes

eklinde sıralanmıştır.

# Recommendations and Conclusion

Program çalıştırılırken boyut indirgenme sayısının 5 olması idealdir ve resimlerde gürültü temizlemeside yapılmalıdır. Sınıflandırma algoritmalarında ise Naive Bayes diğer sınıflandırma algoritmalarına göre daha düşük sonuç vermektedir bu yüzden diğer sınıflandırma algoritmalarına yoğunlaşarak daha iyi sonuçlar elde edilebilir. Toplam resim sayısı 288 dir daha fazla aynı kalitede resim olması durumunda daha iyi sonuçlar elde edilirdi. Projede görüldüğü gibi her zaman fazla parameter değeri daha iyi sonuç getirmeyebilir KNN ‘ de olduğu gibi yada Random Forest ‘daki gibi parameter değeri büyüdükçe daha iyi sonuçlar elde edilebilir.

# References

<Anirudh Gokulaprasad ><22.05.2022> <”Bears Fastai 2021” Veri Seti >

<https://www.kaggle.com/datasets/anirudhg15/bears-fastai-2021>

# Appendices

<Naive Bayes Kullanımı >

<https://scikit-learn.org/stable/modules/naive\_bayes.html>

<KNeighborsClassifier Kullanımı>

<https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>

<Random Forest Kullanımı>

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

<SVC Kullanımı>

<http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>

<Çapraz Doğrulama Kullanımı>

<https://scikit-learn.org/stable/modules/generated/sklearn.model\_selection.KFold.html>

<ResimlerdekiGürültüyü Temizleme>

<https://docs.opencv.org/4.x/d1/d79/group\_\_photo\_\_denoise.html>

<PCA Dönüşümü>

<https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html>